

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БУДІВНИЦТВА І
АРХІТЕКТУРИ

М.І. Цюцюра, К.І. Київська

**СУЧАСНІ МЕТОДОЛОГІЇ ПРОЕКТУВАННЯ ТА РОЗРОБКА
ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

Методичні вказівки до виконання індивідуальної роботи

Київ, 2020

ЗМІСТ

| | |
|--|-----------|
| 1. Тематика індивідуальних робіт | 3 |
| 2. Структура індивідуального проекту | 4 |
| 3. Правила оформлення індивідуальної роботи | 20 |
| 4. Література | 22 |

1. ТЕМАТИКА КУРСОВИХ ПРОЕКТІВ

1. Інженерний калькулятор.
2. Електронний годинник з будильником.
3. Годинники зі стрілками.
4. Зображення графіків різних функцій.
5. Висновок діаграм табличних значень.
6. Програма тестування.
7. Програма анкетування.
8. Записна книжка.
9. Щоденник.
10. Планування робочого часу.

2. СТРУКТУРА індивідуального проекту

Орієнтовна структура індивідуального проекту:

- Титульний аркуш
 - зміст
 - Вступ
 - 1. Розробка ескізного і технічного проекту програми
 - 1.1. Призначення і область застосування
 - 1.2. Технічні характеристики
 - 2. Розробка робочого проекту
 - 2.1 Розробка програми
 - 2.2 Специфікація програми
 - 2.3 Текст програми
 - 2.4 Опис програми
 - 2.5 Тестування програми
 - 3. Оцінка програми Висновки
- Список використаної літератури
Програми

Зміст.

У ньому міститься назва глав і параграфів із зазначенням сторінок.

Вступ.

Під введення обґрунтовується актуальність обраної теми, її важливість. Формуються цілі і завдання курсового проекту. Бажано, щоб було відображено значення розглянутих питань, крім цього у введенні повинна обґрунтовуватися структура курсового проекту.

1. Розробка ескізного і технічного проекту програми.

Даний параграф складається з:

1.1. Призначення і область застосування.

В даному розділі вказують призначення програми і коротку характеристику області застосування програми.

Наприклад: Дана програма є розважальною ігровою програмою. Областю застосування даної програми є організація дозвілля людини, що працює з комп'ютером. Так як, поставлена задача, розробити додаток під операційну систему Windows, то використовуватися програма може тільки в середовищі 32 бітної операційної системи Windows 9x / ME / NT / 2000 / XP.

1.2. Технічні характеристики.

Даний пункт повинен містити наступні підпункти:

1.2.1. Постановка задачі.

Тут дається точний опис вихідних даних, умов завдання і цілей її рішення. На цьому етапі умови задачі, записані у формі різних словесних описів, необхідно висловити за допомогою моделі предметної області (МПО). У цьому підпункті можуть бути описані основні прийоми програмування і типи даних, що використовуються при вирішенні аналогічних завдань. Далі описуються можливі шляхи вирішення завдання із зазначенням їх достоїнств і недоліків. Вибирається і обґрунтовується метод вирішення задачі.

1.2.2. Опис алгоритму.

Тут дається узагальнене словесний опис алгоритму вирішення поставленого завдання, викладаються основні вимоги до алгоритму та шляхи їх реалізації. Наводиться схема алгоритму, що складається з укрупнених модулів. Дається пояснення призначення і складу кожного модуля.

На наступному етапі кожен модуль деталізується.

наприклад: Загальний не деталізований алгоритм виконання даної програми можна представити у вигляді блок схеми зображеної на рис. 1.

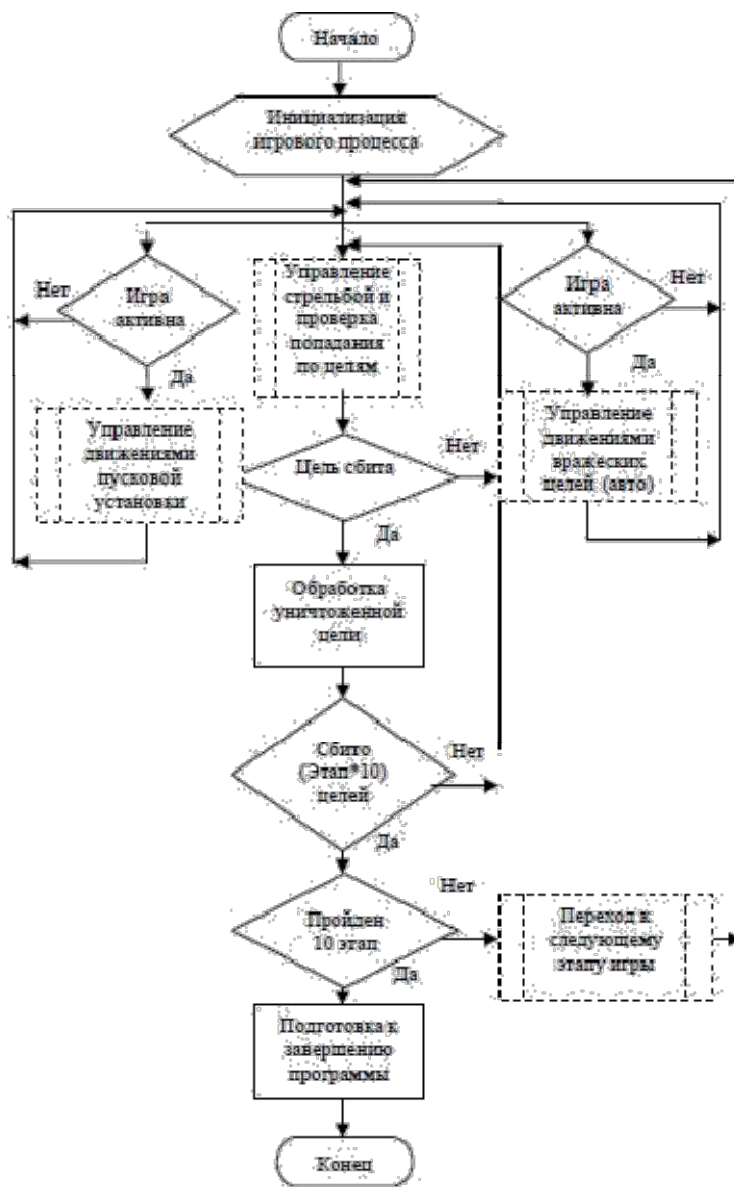


Рис. 1. Общая блок-схема игры «Сбей самолет».

Словесно цю блок-схему можна описати як наступну послідовність дій:

1. Початок;
2. Ініціалізація ігрового процесу (підготовка даних, параметрів і т.д.);
3. Активація трьох паралельних процесів управління;
 - 3.1. Управління рухами пускової установки (за участю користувача);
 - 3.1.1. У разі надходження команди переміщення пускової установки, пускова установка переміщається відповідно до командою і повертає управління з пунктом 3;
 - 3.2. Управління стрільбою по літаках і перевірка попадання по ним;
 - 3.2.1. У разі надходження команди на постріл, відбувається запуск об'єкта «ракета», яка рухається вертикально, поки не вийде за межі видимої області, або поки не потрапить в ціль. Якщо мета не уражена,

управління повертається пункту 3. У разі якщо мета вражена, управління передається пункту 4;

3.3. Управління рухами ворожих цілей (в автоматичному режимі)

3.3.1. Автономний процес управління рухом літаків контролює переміщення цілей по екрану. У разі виходу мети за межі видимої області реалізується «розворот» цілі і її рух в зворотному напрямку. Тут же розраховуються швидкості руху і кількість відображуваних цілей. По завершенні процес передає управління пункту 3. І так далі.

В загальний алгоритм можна виділити основні підзадачі:

1. Управління рухами ворожих цілей.
2. Управління стрільбою, перевірка попадання по цілях.
3. Управління рухами пускової установки.
4. Перехід до наступного етапу гри.

Оскільки найбільш важливими і складно реалізованими частинами програми є саме ці процедури, було прийнято рішення зобразити їх більш деталізовані алгоритми у вигляді блок-схем.

Тобто тут, вибрані модулі деталізуються.

Для оформлення даного підпункту використовується ГОСТ 19.701-90 ЕСПД. Схеми алгоритмів, програм, даних і систем. Умовні позначення і правила виконання.

1.2.3. Організація вхідних і вихідних даних.

Тут міститься опис і обґрунтування вибору методу організації вхідних та вихідних даних.

Наприклад: У програмі були використані три типи вхідних даних:

1. Графічні, представлені у вигляді файлів у форматі bmp.
2. Звукові дані, представлені у вигляді файлів у форматі wav.
3. Відео дані, представлені у вигляді файлів у форматі avi. Існує два способи введення цих даних в програму:
 - пряме завантаження в режимі розробки форм;
 - програмна завантаження даних з файлу в відповідний компонент в режимі роботи програми.

Вихідні дані представляють собою динамічно змінну графічну інформацію, виведену на дисплей ЕОМ.

1.2.4. Вибір складу технічних і програмних засобів.

У цьому підпункті на підставі розробленого алгоритму робиться висновок про необхідність використання тієї чи іншої мови програмування. Перераховуються гідності обраної середовища програмування. Визначаються технічні засоби, необхідні для оптимальної роботи майбутньої програми.

Наприклад: Виходячи з цілей курсової роботи, потрібно створити програму, що працює під ОС Windows. У зв'язку з тим, що розроблений алгоритм має на увазі використання графічних можливостей, звуку,

анімації та паралельних процесів взаємодії з користувачем, було прийнято рішення використовувати інтегроване середовище розробки програм Borland Delphi 6. Дане середовище розробки дозволяє досить швидко створювати додатки для Windows.

Технічні засоби, достатні для оптимальної роботи програми:

- Windows 9x / ME / NT / 2000 / XP.
- Відео пам'ять не менше 1Mb. Прискорювач графіки не потрібно.
- Оперативної пам'яті не менше 16 Mb.
- Процесор Intel Pentium 166 і вище.
- Обсяг вільного місця на жорсткому диску не менше 500 Mb.
- Маніпулятори: клавіатура і миша.

2. Розробка робочого проекту.

Даний параграф складається з:

2.1. Розробка програми.

Проект Windows-вікна повинен бути представлений у вигляді графічної схеми, на якій розташовані всі візуальні і невізуальні компоненти, що розробляється інтерфейсу. Компоненти на схемі повинні бути пронумеровані. Після схеми наводиться розшифрування зображених на схемі компонентів: назва пі ім'я компонента, призначення в програмі, подія на яке даний компонент відгукується.

Інтерфейс з користувачем (всі види взаємодії з користувачем: інтерактивний режим, обробка помилок, автономна робота).

- Визначити вимоги до інтерфейсу користувача.
- Загальні специфікації для користувача функцій.
- Опис основних запитів і алгоритмів обробки даних. Вимоги до графічного інтерфейсу:
 - Утримувати звичні і зрозумілі користувачеві пункти меню, що відповідають повноваженням обробки.
 - Орієнтуватися на користувача, який спілкується з програмою на зовнішньому рівні взаємодії.
 - Задовольняти правилу "шести" - в одну лінійку меню включати не більше 6 понять, кожне з яких містить не більше 6 опцій.
 - Зберігати стандартизоване призначення та місце розташування на екрані графічних об'єктів.

При створенні програми в системі Delphi, до них пред'являються такі вимоги:

- багатовіконне (не менше 2-х вікон);
- наявність меню;
- наявність спливаючих підказок;
- захист від некоректного введення даних;
- обробка виняткових ситуацій;
- наявність відомостей про програму;
- наявність відомостей про автора.

Наприклад: Як уже зазначалося, для розробки програми гри "Збий літак" використовується середу візуального програмування Delphi. В даному середовищі був створений проект з назвою HitThePlane.dpr. Даний проект складається з одного програмного модуля і однієї форми: Form1: (див. Рис. 2)



Головна форма являє собою стандартне вікно класу TForm і містить наступні необхідні для функціонування програми компоненти, розташовані нижче, в порядку їх розміщення:

1. Form1: TForm - Основна форма проекту. Контейнер всіх об'єктів.
2. StartTimer: TTimer - Таймер, який активує ігровий процес.
3. EnemyTimer: TTimer - Таймер для керування рухом літаків.
4. FireTimer: TTimer - Таймер для контролю стрільби і руху ракети.
5. Music: TMediaPlayer - Програвач фонові музики.

І так далі.

При написанні програми не слід забувати про гарному стилі програмування. Після заголовка процедури або функції записується коментар, що містить пояснювальний текст, а саме: призначення підпрограми; перелік і призначення параметрів; їх тип.

2.2. Специфікація програми.

В даному підпункті наводиться точна назва програми і її склад. Оформляється відповідно до ГОСТ 19.202-78 ЕСПД.

Наприклад: Здійснених файл програми «Збий Літак» має назву Hit The Plane.exe і розташований в каталозі \ Збий Літак.

Крім виконуваного файлу в проект включені наступні файли (Табл. 2):

Таблица 2

Таблица файлов входят в проект

| Наименование | Обозначение | Примечание |
|-----------------|------------------------------|------------------------------|
| HitThePlane.dpr | Файл проекта | Связывает все файлы |
| HitThePlane.cfg | Файл конфигурации | Конфигурация |
| HitThePlane.dof | Файл параметров проекта | Текущие установки |
| HitThePlane.res | Файл ресурсов | Содержит пиктограммы и т.д. |
| Unit1.pas | Программный модуль для Form1 | Определяет функции формы |
| Unit1.dcu | Объектный файл для Unit1.pas | Откомпилированная версия |
| Unit1.dfm | Файл формы №1 | Список компонентов |
| IMAGES\ | Папка с графическими файлами | Следующие 10 файлов |
| beginning.bmp | Файл рисунка | Заставка |
| victory.bmp | Файл рисунка | Заставка |
| theend.bmp | Файл рисунка | Заставка |
| Back.bmp | Файл рисунка | Задний фон |
| plane1.bmp | Файл рисунка | Самолет |
| plane2.bmp | Файл рисунка | Самолет |
| jeep.avi | Файл анимации | Анимация запуска |
| SOUND\ | Папка со звуковыми файлами | Следующие 9 файлов |
| BackMusic.mid | Файл полифонии | Фоновая музыка (формат MIDI) |
| Alarm.wav | Звуковой файл | Звук сирены |
| Boom.wav | Звуковой файл | Звук взрыва |
| Start.wav | Звуковой файл | Звук запуска двигателя |

Наводится полный листинг программы разом с комментариями.

например:

```
unit Unit1;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,  
Forms, Dialogs, ExtCtrls,
```

```
StdCtrls, ComCtrls, MPlayer, mmsystem, Menus,  
Buttons; type
```

```
TForm1 = class (TForm)
```

```
BackPanel: TPanel;
```

```
Back: TImage;
```

```
EnemyTimer: TTimer;
```

```
Panel1: TImage;
```

Plane2: TImage;
Enemy4: TImage;
Enemy5: TImage;
Enemy3: TImage;
Enemy2: TImage;
Enemy1: TImage;
Rocket: TPanel;
FireTimer: TTimer;
LedPanel: TPanel;
PtsLed: TLabel;

2.4. Опис програми.

Даний підпункт оформляється згідно з ГОСТ 19.402-78 ЕСПД і повинен містити такі підрозділи:

- Загальні відомості - повинні бути вказані: позначення і найменування програми, програмне забезпечення, необхідне для функціонування програми, мови програмування, на яких написана програма.

- Функціональне призначення - повинні бути вказані класи розв'язуваних завдань і / або призначення програми і відомості про функціональні обмеження на застосування.

- Використовувані технічні засоби - повинні бути вказані типи ЕОМ і пристроїв, які використовуються при роботі програми.

- Виклик і завантаження - повинні бути зазначені спосіб виклику програми з відповідного носія даних.

- Вхідні дані - повинні бути вказані: характер, організація і попередня підготовка вхідних даних, формат, опис і спосіб кодування вхідних даних.

- Вихідні дані - повинні бути вказані: характер, організація і попередня підготовка вихідних даних, формат, опис і спосіб кодування вихідних даних.

Окремі розділи можна об'єднувати. Деякі пункти цього розділу повторюють розділи технічного проекту. Такі повторення передбачені ГОСТом, так як на етапі робочого проекту виникають деякі доповнення або зміни в складі технічних засобів. Тут наводяться більш конкретні і точні дані.

наприклад:

Загальні відомості.

Розроблене додаток є розважальною ігровою програмою і називається «Збий Літак». Програма розроблена в середовищі Borland Delphi 6 на мові Object Pascal.

Функціональне призначення.

Програма призначена для розваги і організації дозвілля людей, що працюють з ЕОМ.

Використовувані технічні засоби.

Для стабільної роботи даної програми потрібні такі технічні та програмні засоби:

- Windows 9x / ME / NT / 2000 / XP.
- Відео пам'ять не менше 1Mb. Прискорювач графіки не потрібно.
- Оперативної пам'яті не менше 16 Mb.
- Процесор Intel Pentium 166 і вище.
- Обсяг вільного місця на жорсткому диску не менше 500 Mb.
- Маніпулятори: клавіатура і миша.

Виклик і завантаження.

Програма запускається на виконання або подвійним клацанням лівої кнопки миші по файлу HitThePlane.exe в провіднику Windows, або натисканням правої кнопки миші по вказаного файлу і вибором пункту меню "Відкрити".

Вхідні дані.

У програмі були використані три типи вхідних даних:

1. Графічні, представлені у вигляді файлів у форматі bmp.
2. Звукові дані, представлені у вигляді файлів у форматі wav.
3. Відео дані, представлені у вигляді файлів у форматі avi.

Існує два способи введення цих даних в програму:

1. Пряме завантаження в режимі розробки форм;
2. Програмна завантаження даних з файлу в відповідний компонент в режимі роботи програми.

Більш наочно вхідні дані відображені в табл. 3.

Таблиця вхідних даних і їх файлів

| Файл | Формат данных | Описание дан |
|---------------|------------------------------------|-----------------------------|
| IMAGES\ | Папка с графическими файлами | Следующие 10 файлов на |
| beginning.bmp | Файл рисунка | Заставка нача |
| victory.bmp | Файл рисунка | Заставка побе |
| theend.bmp | Файл рисунка | Заставка фина |
| Back.bmp | Файл рисунка | Задний фон |
| plane1.bmp | Файл рисунка | Самолет впра |
| plane2.bmp | Файл рисунка | Самолет влен |
| jeep.avi | Файл анимации | Анимация пусковой |
| SOUND\ | Папка со звуковыми файлами | Следующие 9 файлов на |
| BackMusic.mid | Файл полифонии | Фоновая музы (формат MID |
| Alarm.wav | Звуковой файл | Звук сирень |
| Boom.wav | Звуковой файл | Звук взрыва |
| Start.wav | Звуковой файл | Звук запуска двигате |
| Engine.WAV | Звуковой файл | Звук работы двигате |
| Plane.wav | Звуковой файл | Звук самолет |

2.5. Тестування програми.

Коротко описується середовище програмування. Проводяться основні команди, що виконуються при введенні і редагуванні програми, команди запису програми на диск, читання з диска. Перераховуються і описуються засоби налагодження.

Наприклад: При розробці програми використовувалася середовище програмування Borland Delphi 6. Це середовище являє собою набір засобів швидкої розробки і налагодження додатків для Windows.

При розробці програми застосовувався візуальний редактор форм, редактор програмного коду, палітра компонентів і інспектор об'єктів.

Основними операціями в процесі розробки були: створення форм, розміщення компонентів, редагування властивостей цих компонентів, формування візуального призначеного для користувача інтерфейсу, написання коду програми для обробки подій, збереження і компіляція проекту.

Крім зазначених операцій застосовувалися додаткові операції для налагодження створеного додатки. Для цього в складі Delphi 6 є інтегральний відладчик, який дозволяє виявляти і виправляти помилки як на стадії написання коду, так і на стадії налагодження працюючої програми.

3. Оценка программы.

В данном параграфе проводится оценка созданного программного продукта.

Затраты на создание программного продукта складываются из расходов по оплате труда разработчика программы и расходов по оплате машинного времени при отладке программы.

$$Z_{сшт} = Z_{сшт}^{ЗП} + Z_{сшт}^{МЕ} + Z_{общ}.$$

где

$Z_{сшт}$ - затраты на создание программного продукта;

$Z_{сшт}^{ЗП}$ - затраты на оплату труда разработчика программы;

$Z_{сшт}^{МЕ}$ - затраты на оплату машинного времени;

$Z_{общ}$ - общие затраты.

Расходы на оплату труда разработчика программы.

Расходы на оплату труда разработчика программы определяются путем умножения трудоёмкости создания программного продукта на среднюю часовую оплату программиста (с учётом коэффициента отчислений на социальные нужды).

$$Z_{сшт}^{ЗП} = t \cdot T_{час}.$$

Расчёт трудоёмкости создания программного продукта.

Трудоёмкость разработки программного продукта можно определить следующим образом:

$$T = t_0 + t_A + t_B + t_B + t_D + t_{от}.$$

где

t_0 - затраты труда на подготовку описания задачи;

t_A - затраты труда на разработку алгоритма решения задачи;

t_B - затраты труда на разработку блок-схемы алгоритма решения задачи;

t_B - затраты труда на составление программы по готовой блок-схеме;

t_D - затраты труда на подготовку документации задачи;

$t_{от}$ - затраты труда на отладку программы на ЭВМ при комплексной отладке задачи;

Составляющие затрат, в свою очередь можно вычислить через условное число операторов Q . Например, в нашем случае число операторов в отлаженной программе $Q = 6327$.

Расчёт затрат труда на подготовку описания.

Оценить затраты труда на подготовку описания задачи не возможно, т.к. это связано с творческим характером работы, вместо этого оценим затраты труда на изучение описания задачи с учётом уточнения описания и квалификации программиста определяются:

$$t_0 = Q \cdot B / (75 \dots 85 \cdot K),$$

где

B - коэффициент увеличения затрат труда вследствие недостаточного описания задачи уточнений и некоторой недоработки, $B = 1, 2, \dots, 5$;

K - коэффициент квалификации разработчика, для работающих до 2 лет $K = 0, 8$;

В связи с тем, что при изучении описания данной задачи потребовалось много уточнений и доработок в описании коэффициент B принимаем равным 4.

Таким образом, получим:

$$t_{II} = 4000 \cdot 4 / (80 \cdot 0,8) = 250 (\text{чел} / \text{час}).$$

Расчёт затрат труда на разработку алгоритма.

Затраты труда на разработку алгоритма решения задачи:

$$t_A = Q / (60 \dots 75 \cdot K) = 4000 / (70 \cdot 0,8) = 71,43 (\text{чел} / \text{час}).$$

Расчёт затрат труда на разработку блок-схемы.

Затраты труда на разработку блок-схемы алгоритма решения задачи вычислим следующим образом:

$$t_B = Q / (60 \dots 75 \cdot K) = 4000 / (70 \cdot 0,8) = 71,43 (\text{чел} / \text{час}).$$

Расчёт затрат труда на составление программы.

Затраты труда на составление программы по готовой блок-схеме вычислим по формуле:

$$t_D = Q / (60 \dots 75 \cdot K) = 4000 / (70 \cdot 0,8) = 71,43 (\text{чел} / \text{час}).$$

Расчёт затрат труда на отладку программы.

Затраты труда на отладку программы на ЭВМ при комплексной отладке задачи:

$$t_{OT} = 1,5 \cdot t_{OT}^A,$$

где

t_{OT}^A - затраты труда на отладку программы на ЭВМ при автономной отладке одной задачи;

$$t_{OT}^A = Q / (40 \dots 50 \cdot K) = 4000 / (45 \cdot 0,8) = 111,11 (\text{чел} / \text{час}).$$

Отсюда

$$t_{OT} = 1,5 \cdot 111,11 = 166,67 (\text{чел} / \text{час}).$$

Расчёт затрат труда на подготовку документации.

Затраты труда на подготовку документации по задаче определяются:

$$t_D = t_{DP} + t_{DO},$$

где

t_{DP} - затраты труда на подготовку материалов в рукописи;

t_{DO} - затраты на редактирование, печать и оформление документации.

$$t_{DP} = Q / (150 \dots 200 \cdot K) = 4000 / (200 \cdot 0,8) = 25 (\text{чел} / \text{час});$$

$$t_{DO} = 0,75 \cdot t_{DP} = 0,75 \cdot 25 = 18,75 (\text{чел} / \text{час});$$

Отсюда

$$t_D = 18,75 + 25 = 43,75 (\text{чел} / \text{час}).$$

Итак, общую трудоёмкость программного продукта можем рассчитать:

$$t = 250 + 71,43 + 71,43 + 71,43 + 43,75 + 166,67 = 674,71(\text{чел./час}).$$

Расчёт средней зарплаты программиста.

Средняя зарплата программиста в современных рыночных условиях может варьироваться в широком диапазоне. Для примера расчёта возьмём среднюю часовую оплату труда, которая составляет $T_{\text{час}} = 20 \text{ руб./час}$, что составляет 3520 руб./мес. при 8-ми часовом рабочем дне и 5-ти дневной рабочей неделе. Вы должны взять примерную зарплату на данный момент в городе.

Затраты на оплату труда программиста состоят из зарплаты программиста и отчислений на социальные нужды. Отчисления на социальные нужды включают в себя:

- пенсионный фонд (29%),
- медстрах (3,6%),
- соцстрах (5,4%),
- фонд занятости (1,5%),
- сбор на образование (1%).

Итого отчисления на социальные нужды составляют 40,5%. Отсюда затраты на оплату труда программиста составляют:

$$Z_{\text{спт}}^{\text{спт}} = 674,71 \cdot 20 \cdot 1,405 = 18959,35 \text{ руб.}$$

Затраты на оплату машинного времени.

Затраты на оплату машинного времени при отладке программы определяются путём умножения фактического времени отладки программы на цену машино-часа арендного времени:

$$Z_{\text{спт}}^{\text{спт}} = C_{\text{час}} \cdot t_{\text{ЭВМ}},$$

где

$C_{\text{час}}$ - цена машино-часа арендного времени, руб./час;

$t_{\text{ЭВМ}}$ - фактическое время отладки программы на ЭВМ;

Расчёт фактического времени отладки.

Фактическое время отладки вычислим по формуле:

$$t_{\text{ЭВМ}} = t_{\text{п}} + t_{\text{до}} + t_{\text{от}};$$

$$t_{\text{ЭВМ}} = 71,43 + 18,75 + 166,67 = 256,85 \text{ часа.}$$

Расчёт цены машино-часа.

Цену машино-часа найдём по формуле:

$$C_{\text{час}} = Z_{\text{ЭВМ}} / T_{\text{ЭВМ}},$$

где

$Z_{\text{ЭВМ}}$ - полные затраты на эксплуатацию ЭВМ в течении года;

$T_{\text{ЭВМ}}$ - действительный годовой фонд времени ЭВМ, час/год.

Расчёт годового фонда времени работы ПЭВМ IBM PC AT.

Общее количество дней в году - 365.

Число праздничных и выходных дней - 119.

Время простоя в профилактических работах определяется как еженедельная профилактика по 4 часа.

Итого годовой фонд рабочего времени ПЭВМ составляет:

$$T_{ЭВМ} = 8 \cdot (365 - 119) - 52 \cdot 4 = 1760 \text{ часа.}$$

Расчёт полных затрат на эксплуатацию ЭВМ.

Полные затраты на эксплуатацию ЭВМ можно определить по формуле:

$$Z_{ЭВМ} = (Z_{ЗП} + Z_{АМ} + Z_{ЭП} + Z_{ЕМ} + Z_{ТР} + Z_{ПР}),$$

где

$Z_{ЗП}$ - годовые издержки на заработную плату обслуживающего персонала, руб/год;

$Z_{АМ}$ - годовые издержки на амортизацию, руб/год;

$Z_{ЭП}$ - годовые издержки на электроэнергию, потребляемую ЭВМ, руб/год;

$Z_{ЕМ}$ - годовые издержки на вспомогательные материалы, руб/год;

$Z_{ТР}$ - затраты на текущий ремонт компьютера, руб/год;

$Z_{ПР}$ - годовые издержки на прочие и накладные расходы, руб/год;

Амортизационные отчисления.

Сумма годовых амортизационных отчислений определяется по формуле:

$$Z_{АМ} = C_{БАЛ} \cdot H_{АМ},$$

где

$C_{БАЛ}$ - балансовая стоимость компьютера, руб/шт.;

$H_{АМ}$ - норма амортизации, %.

Согласно постановления совета министров СССР от 22 октября 1990 года № 1072 «О единых нормах амортизационных отчислений на полное восстановление основных фондов народного хозяйства СССР» $H_{АМ} = 12,5\%$.

Балансовая стоимость ПЭВМ включает отпускную цену, расходы на транспортировку, монтаж оборудования и его наладку:

$$C_{БАЛ} = C_{РЫН} + Z_{УСТ},$$

где

$C_{РЫН}$ - рыночная стоимость компьютера, руб/шт.;

$Z_{УСТ}$ - затраты на доставку и установку компьютера, руб/шт.;

Компьютер, на котором велась работа, был приобретен по цене $C_{РЫН} = 9000 \text{ руб.}$, (вставить свои данные) затраты на установку и наладку составили примерно 10% от стоимости компьютера:

$$Z_{УСТ} = 10\% \cdot C_{РЫН} = 0,1 \cdot 9000 = 900 \text{ руб.}$$

Отсюда

$$C_{БАЛ} = 9000 + 900 = 9900 \text{ руб / шт.}$$

а

$$Z_{АМ} = 9900 \cdot 0,125 = 1237,5 \text{ руб / год.}$$

Расчёт затрат на электроэнергию.

Стоимость электроэнергии, потребляемой за год, определяется по формуле:

$$Z_{ЭП} = P_{ЭЛ} \cdot T_{ЭВМ} \cdot C_{ЭЛ} \cdot A,$$

где

$T_{ЭВМ}$ - суммарная мощность ЭВМ;

$C_{ЭЛ}$ - стоимость $1 \text{ кВт} \cdot \text{ч}$ электроэнергии;

A - коэффициент интенсивного использования мощности машины.

Согласно техническому паспорту ЭВМ $P_{ЭВМ} = 0,22 \text{ кВт}$, стоимость $1 \text{ кВт} \cdot \text{ч}$ электроэнергии для предприятий $C_{эл} = 0,548 \text{ руб.}$, интенсивность использования машины $A = 0,98$.

Тогда расчётное значение затрат на электроэнергию:

$$Z_{эл} = 0,22 \cdot 1760 \cdot 0,548 \cdot 0,98 = 207,94 \text{ руб.}$$

Расчёт затрат на текущий ремонт.

Затраты на текущий и профилактический ремонт принимаются равными 5% от стоимости ЭВМ:

$$Z_{ТР} = 0,05 \cdot C_{ЭВМ} = 0,05 \cdot 9900 = 495 \text{ руб.}$$

Расчёт затрат на вспомогательные материалы.

Затраты на материалы, необходимые для обеспечения нормальной работы ПЭВМ составляют около 1% от стоимости ЭВМ:

$$Z_{ЕМ} = 0,01 \cdot 9900 = 99 \text{ руб.}$$

Прочие затраты по эксплуатации ПЭВМ.

Прочие косвенные затраты, связанные с эксплуатацией ПЭВМ, состоят из амортизационных отчислений на здания, стоимости услуг сторонних организаций и составляют 5% от стоимости ЭВМ:

$$Z_{ПР} = 0,05 \cdot 9900 = 495 \text{ руб.}$$

Головные издержки на заработную плату обслуживающего персонала.

Издержки на заработную плату обслуживающего персонала складываются из основной заработной платы, дополнительной и отчислений на заработную плату:

$$Z_{ЗП} = Z_{ЗП}^{осн} + Z_{ЗП}^{доп} + Z_{ЗП}^{отч}$$

Сумма основной заработной платы определяется исходя из общей численности работающих в штате:

$$Z_{ЗП}^{осн} = 12 \cdot \sum Z_{окл}^i$$

где

$Z_{окл}^i$ - тарифная ставка i -го работника в месяц, руб.;

В штат обслуживающего персонала должны входить инженер-электронщик с месячным окладом 1000 руб. и электрослесарь с окладом 500 руб.

Тогда, учитывая, что данный персонал обслуживает 10 машин, имеем издержки на основную заработную плату обслуживающего персонала составят:

$$Z_{ЗП}^{осн} = 12 \cdot (1000 + 500) / 10 = 1800 \text{ руб.}$$

Сумма дополнительной заработной платы составляет 60% от основной заработной платы:

$$Z_{ЗП}^{доп} = 0,6 \cdot 1800 = 1080 \text{ руб.}$$

Сумма отчислений на социальные нужды составляет 40,5% от суммы дополнительной и основной заработных плат:

$$Z_{\text{шт}}^{\text{отч}} = 0,405 \cdot (1800 + 1080) = 1166,4 \text{ руб.}$$

Тогда годовые издержки на заработную плату обслуживающего персонала составят:

$$Z_{\text{пл}} = 1800 + 1080 + 1166,4 = 4046,4 \text{ руб.}$$

Полные затраты на эксплуатацию ЭВМ в течение года составят:

$$Z_{\text{ЭВМ}} = 4046,4 + 1237,5 + 207,94 + 99 + 495 + 495 = 6580,84 \text{ руб.}$$

Тогда цена машино-часа арендуемого времени составит:

$$C_{\text{час}} = 6580,84 / 1760 = 3,74 \text{ руб.}$$

А затраты на оплату машинного времени составят:

$$Z_{\text{смм}}^{\text{МЭ}} = 3,74 \cdot 256,85 = 960,62 \text{ руб.}$$

Расчет общих расходов.

Общие расходы это расходы на освещение, отопление, коммунальные услуги и т.п. Они принимаются равными одной трети основной зарплате разработчика программы, т.е. 2249,03 руб.

Тогда затраты на создание программного продукта составят:

$$Z_{\text{смм}} = 18959,35 + 960,62 + 2249,03 = 22169 \text{ руб.}$$

Можно сделать следующие выводы:

Например. В результате расчета затраты на создание данного программного продукта составили 22169 руб. Данную цифру сложно оценить, так как имеющиеся на рынке подобные продукты слишком специфичны и количество их очень мало. Для снижения затрат можно предложить следующие решения. Как видно, три четверти расходов представляют из себя затраты на оплату труда программиста. Снизить эти затраты можно путем повышения эффективности труда программиста за счет использования более современных ЭВМ для работы, повышения удобства рабочего места и прочих факторов. Также при более бережном и аккуратном отношении к компьютерной технике возможно снизить затраты на ремонт, а так же уменьшить привлечение дополнительного персонала на обслуживание компьютеров.

Заключение.

В этой части студент формулирует основные выводы по теме и рекомендации относительно возможностей практического применения материалов работы. Следует также указать на имеющиеся проблемы и возможные перспективы их решения.

Список использованной литературы.

Перечисляется литература, использованная при выполнении курсовой работы, в том числе и ГОСТы.

Приложения.

В приложение нужно поместить все экранные формы программы, а так же возможно и логотипа программы.

3. ПРАВИЛА ОФОРМЛЕННЯ ІНДИВІДУАЛЬНОЇ РОБОТИ

За обсягом індивідуальні проекти повинні бути не менше 30 сторінок друкованого тексту.

Індивідуальна робота повинна бути здана в друкованому вигляді (на комп'ютері). Текст повинен бути надрукований через 1,5 інтервали шрифтом Times New Roman стандартного розміру (14 шрифт) з дотриманням встановлених розмірів відступу від краю листа:

- ліве поле - 30 мм;
- праве поле - 10 мм;
- верхнє і нижнє поля - по 20 мм.

Титульний лист і зміст не нумеруються, але включені в загальну нумерацію.

Зміст проекту, введення, назва параграфів, висновок, список літератури, додаток починається з нової сторінки. Кожен самостійний розділ нумерується і Озаглавлювати аналогічно тому, як це зроблено в "Змісті". Назви параграфів друкуються симетрично тексту (по центру), 16 шрифтом, жирним. Назви пунктів друкуються з нового рядка малими літерами (крім першої), 14 шрифтом, жирним.

Відстань між заголовком параграфа і основним текстом має дорівнювати трьом (одинарним) міжрядковим інтервалам.

Малюнки (схеми, графіки) повинні бути виконані розбірливо і мати написи, що пояснюють, які розташовані під малюнком (приклади дивіться вище). У текст має бути посилання на кожен малюнок (Наприклад, дані ... см. На рис.1).

При оформленні таблиці слово «Таблиця» пишеться в лівому верхньому кутку (без лапок) і ставиться є порядковий номер. Нумерація може бути наскрізний через всю роботу або за параграфами. Таблиця повинна містити назву (табл. 3 - Таблиця вхідних даних і їх файлів).

При згадці про таблиці в тексті робиться посилання (табл. 1.1). Список літератури повинен містити перелік всіх джерел (НЕ менше 5), використаних при написанні індивідуальної роботи. Джерела наводяться в алфавітному порядку (по першій букві першого слова. В авторських джерелах першим словом вважається прізвище автора).

- всі джерела в переліку нумеруються. Для кожного джерела вказуються: прізвище та ініціали автора (авторів); повна назва книги, статті; назва журналу або збірника статей (для статей);

- назва міста (Москва і Санкт-Петербург - скорочено, відповідно М. і СПб., інші повністю);

- назва видавництва (якщо мається на вхідних даних) - для книг;

- рік видання (для статей - N журналу і рік).

наприклад:

1. Стукотів, С.А. Економічний аналіз і ринок / С.А. Стукотів. // Бухгалтерський облік. -1991.- №3.- С. 106-111.

2. Ткач, В.І. Міжнародна система обліку і звітності / В.І. Ткач. - М.: Фінанси і статистика, 1992. - 528с.

3. Управління за результатами. Пер. з фін. / Под ред. А.Я. Лейтмана. - М.: Прогрес, 1993. - 157с.

Приклади описів електронних ресурсів:

- *Бібліографічні записи Інтернет-ресурсів.*

Досліджено в Росії [Електронний ресурс]: многопредмет. науч. журн. / Моск. фіз.-техн.ін-т. - Електрон. журн. - Долгопрудний: МФТІ, 1998. Режим доступу до журн.: [Http://zhurnal.mipt.rssi.ru](http://zhurnal.mipt.rssi.ru).

Російська державна бібліотека [Електронний ресурс] / Центр інформ. технологій РДБ; ред. Власенко Т.В.; Web-майстер Козлова Н.В. - Електрон. дан. - М.: Рос. держ. б-ка, 1997. Режим доступу: <http://www.rsl.ru>, вільний. - Загл. з екрану. - Яз. рус., англ.

- *Бібліографічні записи збірок електронних ресурсів без загального заголовка:*

Англійська для бізнесменів [Електронний ресурс]; Англійський технічний;

Англійська для читання газет і журналів: [до зб. в цілому]: курс вивчення іноз. яз. Intell. Електрон. дан. і програм. - М.: сміття. Квант, 1994 1997. - 1 електрон., Опт. диск (CD-ROM): зв., цв. -(25 кадр).

- *Бібліографічні записи електронних ресурсів, які складаються з декількох окремих частин (випусків):*

Велика автомобільна енциклопедія [Електронний ресурс]. - Електрон. дан. - М.: Xelana Media Group, 1998..

Крім основного тексту індивідуальна робота може містити додатки. В якості додатків доцільно оформляти код програмного продукту і основні екрани інтерфейсу.

В цьому випадку в основному тексті в необхідних місцях повинні міститися посилання: «см. додаток». Додаток розташовується безпосередньо за списком літератури, на першому аркуші доданих документів у верхньому правому кутку пишеться слово "Додаток" (а потім його порядковий номер) то ж пишеться в останньому рядку змісту.

Кожна програма має мати заголовок, який розташовується по центру. Додатки розташовуються в порядку посиланням на них в основному тексті.

ЛІТЕРАТУРА

1. Вендров А.М. Практикум з проектування програмного забезпечення економічних інформаційних систем. - М.: Фінанси і статистика, 2004.
2. Іванова Г.С. Технологія програмування. - М.: Изд-во МГТУ ім. Н.е. Баумана, 2003.
3. Орлов С.А., Цількер Б.Я. Технології розробки програмного забезпечення. - СПб.: Пітер, 2012.

www.rugost.com

<http://khpi-iip.mipk.kharkiv.edu/library/case/leon/index.html> (Леоненков. Самучитель UML)