Mr. Giansante

C++ Programming

Adventure Game

August 2018

Adventure Game

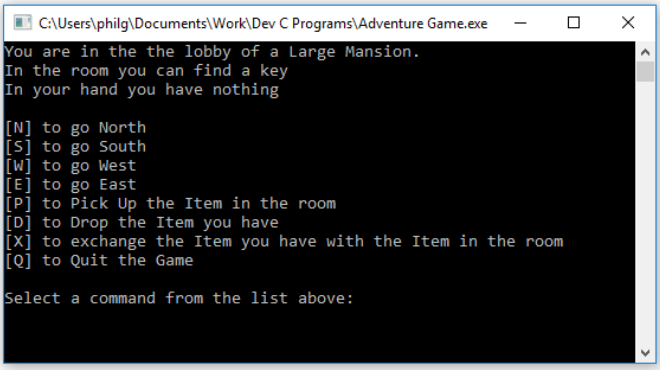
Design a C++ text-based adventure game.
The user should be able to navigate through a world with a minimum of 60 locations.

Examples of worlds include: a large mansion, a castle, the school, a city, a map of North America (50 states, 10 provinces) or even an ancient Egyptian pyramid. The world need not be rectangular.

In addition, the user should be able to pick up and drop objects. To simplify the task, the user should only be able to hold one object at a time and each room can only contain one object. There should be about 10 objects available.

User Interface

The user interface should clearly indicate to the player where they are, the objects available and in which direction they can move.



The screenshot above is simply an example showing all possible options. For example, since there is no object in the player's hand, the "Drop Item" option should not be available.

Designing the Game

Print the "Adventure Game Grid " available on my website. This will help you organize and design your game.

1 Trophy Room	2 Coat Room	3 Main Entrance	4 Men's Washroom	5 Smoking Room
6 West-Wing Hallway	7 West-Wing Hallway	8 Main Hallway	9 East-Wing Hallway	10 East-Wing Hallway
11 Master Bedroom	12 Guest Bedroom	13 Main Hallway	14 Library	15 Reading Room

In the example above, the starting room is the "Main Entrance" (room 3).

From Room 3, the user can go South, in which case they will end up in Room 8, or go West, in which case they will end up in Room 2. The user cannot go North or East.

To store the information, six arrays are required ...

```
string Desc[81];
string Objects[81];
int North[81];
int South[81];
int East[81];
int West[81];

int roomnum = 3;
string CurrentObject = "";
```

Storing the information for each room ...

```
Desc[1] = "Trophy Room";
Objects[1] = "A Trophy";
North[1] = 0;
South[1] = 6;
East[1] = 0;
West[1] = 0;

Desc[2] = "The Coat Room";
Objects[2] = "An Umbrella";
North[2] = 0;
South[2] = 0;
East[2] = 3;
West[2] = 0

Desc[3] = "Main Entrance";
Objects[3] = "";
North[3] = 0;
South[3] = 8;
East[3] = 0;
West[3] = 2;
```

... fill in information for remaining rooms ...

Adventure Game

Start Screen

You can use the website below to generate large ASCII text for your start screen:

```
http://patorjk.com/software/taag/
```

When using this website note that you will need to replace the "\" characters with "\\".

Possible Enhancements to Game

1. **Multiple Floors**

   User can enter a "room" which is an elevator or stairs. They would then have the choice to go "Up" or "Down" (depending on where they are).

2. **Locked Doors**

   The user must find keys to gain access to certain rooms. For example: the blue key opens the blue room.

3. **Timed Game**

   Using a Timer control, you can limit the amount of time that the player has to navigate the game or find certain objects.

4. **Graphical Representation of Rooms**

   Add code to "draw" a graphical representation of each room using ASCII characters.

   Example: If the player can go North and East ...

```
XXXXXX    XXXXXX
X              X
X              X
X
X
X              X
X              X
XXXXXXXXXXXXXXXX
```

   A better to accomplish this is to use ASCII character 219 (a filled in rectangle) instead of X's.

   The code would look something like this ...

```
cout << (char)219 << (char)219 << (char)219 << (char)219 << (char)219 << (char)219 <<"\n";
cout << (char)219 << "            " << (char)210 << "\n";
```

Sample Code

To determine if the "[N] to go North" option should appear ...

```
if(North[roomnum] != 0)
{
     cout << "[N] to go North\n";
}
```

If the user selects the option to "Go North", the following line of code will update the roomnum ...

```
roomnum = North[roomnum];
```

To determine if the "[P] to Pick Up the Item in the room" option should appear ...

```
if(Objects(roomnum) != "" &&
CurrentObject == "")
{
     cout << "[P] to Pick Up the
     Item in the room\n";
}
```

If the user selects the option to "Pick Up Object", the following lined of code will update the corresponding variables ...

```
// Place object in room in player's
// hand and clear the object that
// was in the room

CurrentObject = Objects[roomnum];
Objects[roomnum] = "";
```

The code for the "Drop Object" and "Swap Objects" buttons is similar.

Adventure Game

Learning Outcomes

Ontario

Programming Concepts and Skills
A1. demonstrate the ability to use different data types, including one-dimensional arrays, in computer programs
A2. demonstrate the ability to use control structures and simple algorithms in computer programs
A4. use proper code maintenance techniques and conventions when creating computer programs

Software Development
B1. use a variety of problem-solving strategies to solve different types of problems independently and as part of a team
B3. design algorithms according to specifications
B4. apply a software development life-cycle model to a software development project

Please fill out all the information in this column using a pen before getting this assignment marked.

_____
Name

_____ _____
Date Class

Academic Honesty

The work I am submitting is completely my own creation and has not been copied from anyone else's work. If I have received help on this project, the names of those who have assisted are listed below.

_____
Signature

_____

_____

Pre-Marking

The following people have pre-marked this assignment: (minimum of two)

_____

_____

User Interface

- ❑ Appropriate Title, Author Information, etc.
- ❑ Efficient and Esthetically-Pleasing User Interface
- ❑ Spelling and Grammar are correct

Code / Programming Style

- ❑ Variables are declared and logically named
- ❑ Code is commented where appropriate
- ❑ Code is indented and spaced to show stucture
- ❑ Code is efficient

Work Ethic / Problem Solving Skills

- ❑ Problem Solving Skills are demonstrated
- ❑ Makes Productive Use of Time
- ❑ Only Seeks Help when Necessary

Program-Specific Criteria

- ❑ Initial room information is displayed
- ❑ Object information is displayed in certain rooms
- ❑ North, South, East and West options are displayed when appropriate
- ❑ Drop, Pickup and Swap options are displayed when appropriate
- ❑ Choosing [N], [S], [W] or [E] brings user to a new room and information is updated immediately
- ❑ [D] command works, display is updated immediately
- ❑ [P] command works, display is updated immediately
- ❑ [X] command works, display is updated immediately
- ❑ [Q] command works
- ❑ Rooms follow printout of map provided by student

Teacher Comments

_____

_____

Level 1 - Limited ability to meet standard and limited effectiveness
Level 2 - Some ability demonstrated and moderately effective
Level 3 - Considerable ability demonstrated, considerable clarity or accuracy
Level 4 - Thorough, high degree of skill demonstrated, insightful, highly accurate

Level