Deakin University

Introduction to Programming

OnTrack Submission

Shape Drawing

Submitted By:
Sarah Satish Masih
smasih
2020/03/18 11:29

Tutor:
Jake Renzella

Outcome	Weight
ULO2: Code	♦◇◇◇◇
ULO4: Relate to Outcomes	♦◇◇◇◇

This program helps in understanding how the code is to be written and compiled properly.

March 18, 2020

SIT102 – Introduction to Programming

Answers for 1.2P Shape Drawing

Student Name: Sarah Satish Masih
Student ID: 219447652

Question 1: What are **clear_screen(…), fill_rectangle(…), write_line(…)** examples of? What kind of artefact are they?

Clear_screen(..), fill rectangle(…) and write_line(…) are examples of drawing procedures.

clear_screen(…) is the kind of artefact used to clear a screen when there is some other display on it. Fill_rectangle(…) is a command used to create a rectangle and fill a particular colour in it. Write_line(…) is used to display strings that could contain strings on the output window.

Question 2: What is the name of the statement (action) used to run the code in **clear_screen(…)**, **fill_rectangle(…)**, or **write_line**?

The name of the statement (action) used to run the code is a procedure call. A procedure call is used to call a function that has a set of instructions that can be used to perform a variety of tasks. Clear_screen(), fill_rectangle() and write_line() are procedure calls that call a pre-written code from the library.

Question 3: In what order will the computer run the instructions in your code? Where do these instructions start from?

The instructions of a code will start from the lines written in main() function. It will sequentially follow and carry out the code according to order in which the lines are written, starting on the first and completing its instruction before moving towards the second line of code. If the first line of code in the main() function is calling another function, the computer will call that function and perform the tasks given in that function before returning to main(). In the other function, the computer will also go sequentially by performing the first line of code and then moving to the next.

These instructions start from the main() function.

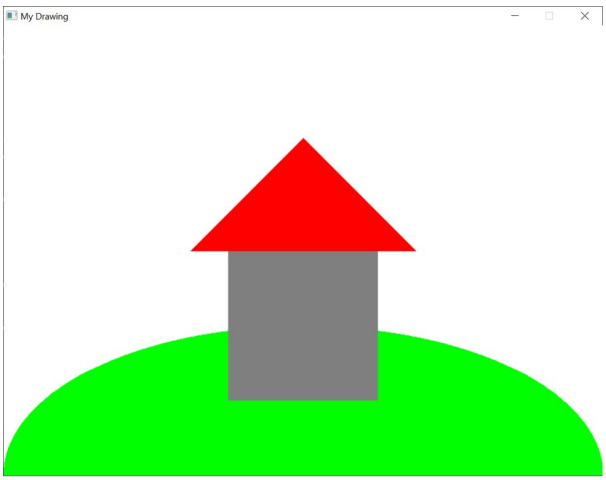
Question 4: Why do we create procedures? What are the advantages of doing this? How will this helps make it easier to build larger programs?

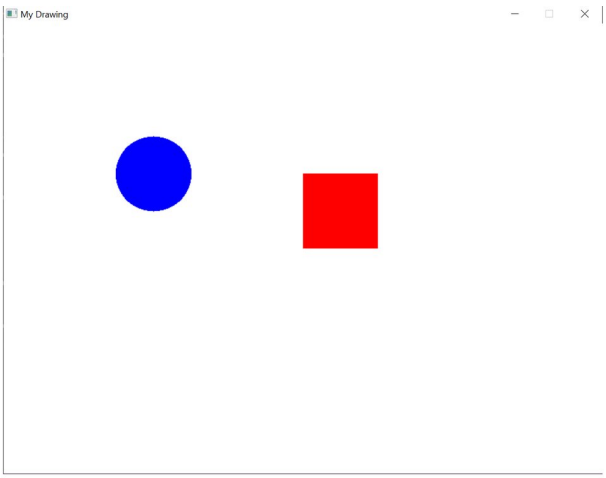
We create procedures so that we can decompose the program into smaller problems that we can solve easily before connecting them in the end so that the entire program is processed smoothly.

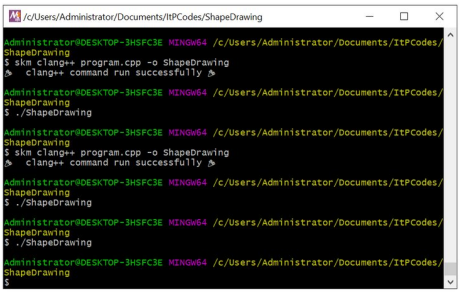
The advantages of doing this are that we can divide the big problem into many small ones and focus on every problem so that it makes it easier to solve and navigate when an error occurs. It also helps in optimising the speed of the computer when a large code is run.

In larger programs, it is very difficult to navigate to find an error or run the large program smoothly when large amounts of data is stored and needs to be processed. To help in the

optimisation of the code and to reduce processing times, we create procedures in larger programs. This helps in grouping several similar lines of code that perform a task into a procedure which the computer will only process when required.


My Drawing


My Drawing


/c/Users/Administrator/Documents/ItPCodes/ShapeDrawing
Administrator@DESKTOP-3HSFC3E MINGW64 /c/Users/Administrator/Documents/ItPCodes/ShapeDrawing
$ skm clang++ program.cpp -o ShapeDrawing
clang++ command run successfully
Administrator@DESKTOP-3HSFC3E MINGW64 /c/Users/Administrator/Documents/ItPCodes/ShapeDrawing
$ ./ShapeDrawing
Administrator@DESKTOP-3HSFC3E MINGW64 /c/Users/Administrator/Documents/ItPCodes/ShapeDrawing
$ skm clang++ program.cpp -o ShapeDrawing
clang++ command run successfully
Administrator@DESKTOP-3HSFC3E MINGW64 /c/Users/Administrator/Documents/ItPCodes/ShapeDrawing
$ ./ShapeDrawing
Administrator@DESKTOP-3HSFC3E MINGW64 /c/Users/Administrator/Documents/ItPCodes/ShapeDrawing
$ ./ShapeDrawing
Administrator@DESKTOP-3HSFC3E MINGW64 /c/Users/Administrator/Documents/ItPCodes/ShapeDrawing
$

```
#include "splashkit.h"


void Draw_House_Scene()
{


    clear_screen(COLOR_WHITE);
    fill_ellipse(COLOR_BRIGHT_GREEN, 0, 400, 800, 400);
    fill_rectangle(COLOR_GRAY,300,300,200,200);
    fill_triangle(COLOR_RED, 250,300,400,150,550,300);
    refresh_screen(60);

}
void Second_Scene()
{

    clear_screen(COLOR_WHITE);
    fill_circle(COLOR_BLUE,200,200,50);
    fill_rectangle(COLOR_RED, 400,200,100,100);
    refresh_screen(60);
}


int main()
{
    open_window("My Drawing", 800,600);
    Draw_House_Scene();
    delay(3000);
    clear_screen(COLOR_BLACK);
    refresh_screen(60);
    delay(3000);

    Second_Scene();
    delay(3000);
    clear_screen(COLOR_BLACK);
    refresh_screen(60);
    delay(3000);

    Draw_House_Scene();
    delay(3000);


    return 0;


}
```