**C++ Programming**

**Sheet 2 – Selection controls (part2)**

**Q1 What is the output of the following statements?**

|  |  |
| --- | --- |
| **if ('R' < '$' && '&' <= '#')**  **cout << "$#";**  **cout << "R&";**  **cout << endl;** | **R&** |
| **if ('4' > '3' || 2 < -10)**  **cout << "1 2 3 4" << endl;**  **cout << "$$" << endl;** | **1 2 3 4**  **$$** |
| **if ("Jack" <= "John" && "Business" >= "Accounting")**  **cout << "Jack Accounting" << endl;**  **cout << "John Business" << endl;** | **Jack Accounting**  **John Business** |
| **int num = 10; //Line 1**  **double temp = 4.5; //Line 2**  **bool found; //Line 3**  **found = (num == 2 \* static\_cast<int>(temp) + 1); //Line 4**  **cout << "The value of found is: " << found << endl; //Line 5** | **The value of found is: 0** |
| **int num = 10; //Line 1**  **double temp = 4.5; //Line 2**  **bool found; //Line 3**  **found = (num == 2 \* static\_cast<int>(temp + 1)); //Line 4**  **cout << "The value of found is: " << found << endl; //Line 5** | **The value of found is: 1** |

**Q2. State whether the following are valid switch statements. If not, explain why. Assume that n and digit are int variables.**

|  |  |
| --- | --- |
| **switch (n <= 2)**  **{**  **case 0:**  **cout << "Draw." << endl;**  **break;**  **case 1:**  **cout << "Win." << endl;**  **break;**  **case 2:**  **cout << "Lose." << endl;**  **break;**  **}** | **valid** |
| **switch (digit / 4)**  **{**  **case 0,**  **case 1:**  **cout << "low." << endl;**  **break;**  **case 1,**  **case 2:**  **cout << "middle." << endl;**  **break;**  **case 3:**  **cout << "high." << endl;**  **}** | **invalid; a case value cannot appear more than once and there should be a colon after a case value.** |
| **switch (n % 6)**  **{**  **case 1:**  **case 2:**  **case 3:**  **case 4:**  **case 5:**  **cout << n;**  **break;**  **case 0:**  **cout << endl;**  **break;**  **}** | **valid** |

|  |  |
| --- | --- |
| **switch (n % 10)**  **{**  **case 2:**  **case 4:**  **case 6:**  **case 8:**  **cout << "Even";**  **break;**  **case 1:**  **case 3:**  **case 5:**  **case 7:**  **cout << "Odd";**  **break;**  **}** | **valid** |

**Q3. In a right triangle, the square of the length of one side is equal to the sum of the squares of the lengths of the other two sides. Write a program that prompts the user to enter the lengths of three sides of a triangle and then outputs a message indicating whether the triangle is a right triangle.**

|  |
| --- |
| **#include <iostream>**  **using namespace std;**  **int main()**  **{**  **double side1, side2, side3;**  **cout << "Enter the lengths of the sides a triangle: ";**  **cin >> side1 >> side2 >> side3;**  **cout << endl;**  **if ((side1 \* side1 == (side2 \* side2 + side3 \* side3)) ||**  **(side2 \* side2 == (side1 \* side1 + side3 \* side3)) ||**  **(side3 \* side3 == (side1 \* side1 + side2 \* side2)))**  **cout << "It is a right angled triangle" << endl;**  **else**  **cout << "It is not a right angled triangle" << endl;**  **return 0;**  **}** |

**Q4.Write a program that calculates and prints the bill for a cellular telephone company, The company offers two types of service: regular and premium. Its rates vary, depending on the type of service. The rates are computed as follows:**

**Regular service: $10.00 plus first 50 minutes are free. Charges for over 50 minutes are $0.20 per minute.**

**Premium service: $25.00 plus:**

**a . For calls made from 6:00 a.m. to 6:00 p.m., the first 75 minutes are free; charges for more than 75 minutes are $0.10 per minute.**

**b. For calls made from 6:00 p.m. to 6:00 a.m., the first 100 minutes are free; charges for more than 100 minutes are $0.05 per minute.**

**Your program should prompt the user to enter an account number, a service code (type char), and the number of minutes the service was used. A service code of r or R means regular service; a service code of p or P means premium service. Treat any other character as an error.**

**Your program should output the account number, type of service, number of minutes the telephone service was used, and the amount due from the user.**

**For the premium service, the customer may be using the service during the day and the night. Therefore, to calculate the bill, you must ask the user to input the number of minutes the service was used during the day and the number of minutes the service was used during the night.**

|  |
| --- |
| **#include <iostream>**  **#include <iomanip>**    **using namespace std;**  **const double REG\_SERV\_CHARGES = 10.00;**  **const int REG\_FREE\_MINUTES = 50;**  **const double REG\_RATE\_OVER\_50 = 0.20;**  **const double PREM\_SERV\_CHARGES = 25.00;**  **const int PREM\_FREE\_DAY\_MINUTES = 75;**  **const double PREM\_DAY\_RATE\_OVER\_75 = 0.10;**  **const int PREM\_FREE\_NIGHT\_MINUTES = 100;**  **const double PREM\_NIGHT\_RATE\_OVER\_100 = 0.05;**  **int main()**  **{**  **int accountNo;**  **char serviceType;**  **int minutesUsed;**  **int minutesUsedPN;**  **double amountDue;**  **cout << "Enter account number: ";**  **cin >> accountNo;**  **cout << endl;**  **cout << "Enter service type: (r or R) for regular, "**  **<< "(p or P) for premium service: " << endl;**  **cin >> serviceType;**  **cout << endl;**  **switch (serviceType)**  **{**  **case 'r':**  **case 'R':**  **cout << "Enter number of minutes service used: ";**  **cin >> minutesUsed;**  **cout << endl;**  **if (minutesUsed <= REG\_FREE\_MINUTES)**  **amountDue = REG\_SERV\_CHARGES;**  **else**  **amountDue = REG\_SERV\_CHARGES**  **+ (minutesUsed - REG\_FREE\_MINUTES)**  **\* REG\_RATE\_OVER\_50;**  **cout << "Account Number: " << accountNo << endl;**  **cout << "Serice Type: Regular" << endl;**  **cout << "Minutes Service Used: " << minutesUsed << endl;**  **cout << "Amount Due: $" << amountDue << endl;**  **break;**  **case 'p':**  **case 'P':**  **cout << "Enter day time minutes used: ";**  **cin >> minutesUsed;**  **cout << endl;**  **cout << "Enter night time minutes used: ";**  **cin >> minutesUsedPN;**  **cout << endl;**  **amountDue = PREM\_SERV\_CHARGES;**  **if (minutesUsed > PREM\_FREE\_DAY\_MINUTES)**  **amountDue = amountDue +**  **(minutesUsed - PREM\_FREE\_DAY\_MINUTES)**  **\* PREM\_DAY\_RATE\_OVER\_75;**  **if (minutesUsedPN > PREM\_FREE\_NIGHT\_MINUTES)**  **amountDue = amountDue +**  **(minutesUsedPN - PREM\_FREE\_NIGHT\_MINUTES)**  **\* PREM\_NIGHT\_RATE\_OVER\_100;**  **cout << "Account Number: " << accountNo << endl;**  **cout << "Serice Type: Premium" << endl;**  **cout << "Minutes Service Used (Day): "**  **<< minutesUsed << endl;**  **cout << "Minutes Service Used (Night): "**  **<< minutesUsedPN << endl;**  **cout << "Amount Due: $" << amountDue << endl;**  **break;**  **default:**  **cout << "Invalid Service Type" << endl;**  **}//end switch**  **return 0;**  **}** |