



**NATIONAL UNIVERSITY OF SCIENCES & TECHNOLOGY**  
**School of Mechanical & Manufacturing Engineering (SMME)**  
**Department of Mechanical Engineering**

Time: 180 minutes

**End Semester Exam**

Maximum Marks: 60

Program: BS Mechanical Engineering

Class: ME-07 (A & B)

Course: EC-102 Computer Systems & Programming

Instructor: Dr. Syed Omer Gilani

Name: \_\_\_\_\_

Section: \_\_\_\_\_

Registration: \_\_\_\_\_

Date: \_\_\_\_\_

**Instructions:**

1. Total time allowed is 180 minutes.
2. Notes and books are not allowed.
3. Calculator is allowed but exchange is prohibited.
4. Question papers must be submitted with answer sheets.
5. Anybody cheating or helping others to cheat will get his/her paper cancelled.
6. Attempt all questions.

## 1 Multiple Choice Questions

1. What will be output if you will compile and execute the following code?

[3]

```
#include <iostream>
using namespace std;
void main(){
    int i=10;
    static int x=i;
    if(x==i)
        cout<<"Equal";
    else if(x>i)
        cout<<"GreaterThan";
    else
        cout<<"LessThan";
}
```

- (a) Equal
- (b) GreaterThan
- (c) LessThan
- (d) Compiler error
- (e) None of above

**ANS** **Option D:** Explanation: static variables are load time entity while auto variables are run time entity. We can not initialize any load time variable by the run time variable. In this example i is run time variable while x is load time variable.

2. What will be output if you will compile and execute the following code?

[3]

```
#include <iostream>
using namespace std;
void main(){
    int i;
    float a=5.2;
    char *ptr;
    ptr=(char *)&a;
    for(i=0;i<=3;i++)
        cout<<*ptr++;
}
```

- (a) 0 0 0 0
- (b) Garbage Garbage Garbage Garbage
- (c) 102 56 -80 32
- (d) 102 102 -90 64
- (e) Compiler error

**ANS** **Option D:**

Answer: (d)

Explanation:

In c float data type is four byte data type while char pointer ptr can point one byte of memory at a time.

Memory representation of float a=5.2



ptr pointer will point first fourth byte then third byte then second byte then first byte.

Content of fourth byte:

Binary value=01100110

Decimal value=  $64+32+4+2=102$

Content of third byte:

Binary value=01100110

Decimal value= $64+32+4+2=102$

Content of second byte:

Binary value=10100110

Decimal value= $-128+32+4+2=-90$

Content of first byte:

Binary value=01000000

Decimal value=64

**Note:** Character pointer treats MSB bit of each byte i.e. left most bit of above figure as sign bit.

3. What will be output if you will compile and execute the following code?

[3]

```
#include <iostream>
#include "string.h"
using namespace std;
void main(){
    cout<<sizeof("string")<<strlen("string");
}
```

(a) 6 6

(d) 7 6

(b) 7 7

(c) 6 7

(e) None of these

**ANS** **Option D** Explanation: Sizeof operator returns the size of string including null character while strlen function returns length of a string excluding null character.

4. What will be output if you will compile and execute the following code?

[3]

```
#include <iostream>
using namespace std;
int extern x;
void main()
{
    cout<<x;
    x=2;
}
```

```
int x=23;
```

- (a) 0  
(b) 2  
(c) 23  
(d) Compiler error  
(e) None of these

**ANS:**

**Option C:** Explanation: extern variables can search the declaration of variable any where in the program.

5. What will be output if you will compile and execute the following code?

[3]

```
#include <iostream>
using namespace std;
int main ()
{
    if(cout<<"c++questions")
        cout<<"I_know_c++";
    else
        cout<<"I_know_c";
    return 0;
}
```

- (a) I\_know\_c  
(b) I\_know\_c++  
(c) c++questionsI\_know\_c++  
(d) c++questionsI\_know\_c

**ANS:**

**Option C** Explanation: Return type of cout function is integer which returns number of character it prints including blank spaces. So cout function inside if condition will return 13. In if condition any non- zero number means true so else part will not execute.

## 2 Write a complete C++ program

6. Write a program to covert decimal number to octal number. Octal number system is base 8 number system which uses the digits from 0 to 7.

**Sample Runs:** Two sample runs are shown below using interactive input in first line and output of your program in second line.

```
>>Enter any decimal number: 25
Equivalent octal number is: 31
```

```
>>Enter any decimal number: 50
Equivalent octal number is: 62
```

[5]

```
#include<stdio.h>

int main() {

    long int decimalNumber,remainder,quotient;
    int octalNumber[100],i=1,j;

    printf("Enter any decimal number: ");
    scanf("%ld",&decimalNumber);

    quotient = decimalNumber;

    while(quotient!=0){
        octalNumber[i++]= quotient % 8;
        quotient = quotient / 8;
    }

    printf("Equivalent octal value of decimal number %d:",decimalNumber);
    for(j = i -1 ;j> 0;j--)
        printf("%d",octalNumber[j]);

    return 0;
}
```

7. Write a program to check if a given number is Armstrong number or not. An Armstrong number is a number such that the sum of its digits raised to the third power is equal to the number itself. For example  $153 = 1^3 + 5^3 + 3^3 = 1 + 125 + 27 = 153$

**Sample Run:** A sample run is shown below using interactive input in first line and output of your program in second line.

>>Enter a number: 370

370 is an Armstrong number

[5]

```
1. Armstrong number in c using for loop

#include<stdio.h>
int main() {
    int num,r,sum=0,temp;

    printf("Enter a number: ");
    scanf("%d",&num);

    for(temp=num;num!=0;num=num/10){
        r=num%10;
        sum=sum+(r*r*r);
    }
    if(sum==temp)
        printf("%d is an Armstrong number",temp);
    else
        printf("%d is not an Armstrong number",temp);

    return 0;
}

Sample output:
Enter a number: 370
370 is an Armstrong number
Logic of Armstrong number in c
```

8. Write a program that can print initials of any name. **Program should only be implemented using char\***

**Sample Run:**

A sample run is shown below using interactive input in first line and output of your program in second line.

>> Enter a string: Robert De Niro

RDN

[5]

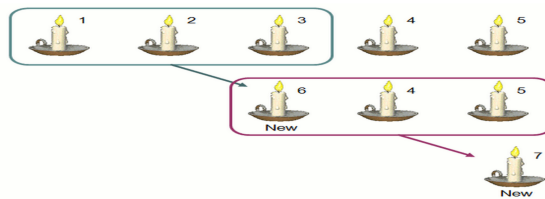
**C code which prints initial of any name**

```
#include<stdio.h>
int main(){
    char str[20];
    int i=0;
    printf("Enter a string: ");
    gets(str);
    printf("%c",*str);
    while(str[i]!='\0'){
        if(str[i]==' '){
            i++;
            printf("%c",*(str+i));
        }
        i++;
    }
    return 0;
}
```

**Sample output:**

```
Enter a string: Robert De Niro
RDN
```

9. Ahmad has  $n$  candles. He burns them one at a time and carefully collects all unburnt residual wax. Using the residual wax of exactly  $k$  (where  $k > 1$ ) candles, he can roll out a new candle. Write a generalized program `candles.cpp` to help Ahmad find out how many candles he can burn in total, given two positive integers  $n$  and  $k$ . The output should print the total number of candles he can burn.



**Sample Runs:** Two sample runs are shown below using interactive input in first line and output of your program in second line

```
>>Enter number of candles and number of residuals to make a new candle: 5 3
Total candles burnt = 7
```

```
>>Enter number of candles and number of residuals to make a new candle: 100 7
Total candles burnt = 116
```

[15]

```
#include <stdio.h>

/* function prototype */
int count_candles(int, int);

int main(void)
{
    int num,          // initial no. of candles
        roll_out_k; // out of k can build one more candles

    printf("Enter number of candles and \n"
           "number of residuals to make a new candle: ");
    scanf("%d %d", &num, &roll_out_k);

    printf("Total candles burnt = %d\n",
           count_candles(num, roll_out_k));

    return 0;
}

// calculates how many candles in total
int count_candles(int candles, int roll_out_num)
{
    int total = candles; // total number of candles to build

    // while still able build more from wax
    while (candles >= roll_out_num)
    {
        // update no. of candles burnt so far
        total += candles / roll_out_num;
        // update remaining candles
        candles = candles/roll_out_num + candles%roll_out_num;
    }

    return total;
}
```

10. Develop a program that plays the game of "Rock, paper, scissors". In this game, two players simultaneously say (or display a hand symbol representing) either "rock", "paper", or "scissors". The winner is the one whose choice dominates the other. The rules are: paper dominates (wraps) rock, rock dominates (breaks) scissors, and scissors dominate (cut) paper.

**Use enumerated types for the choices and for the results.**

**Sample Runs:** Two sample runs are given below.

```
>>> Choose rock (0), paper(1), or scissors (2):
>>> Player1: 1
>>> Player2: 1
You are tied.
```

```
>>> Choose rock (0), paper(1), or scissors (2):
>>> Player1: 2
>>> Player2: 1
Player 1 wins.
```

First define the two enum types Choice and Result. Then declare variables choice1, choice2, and result of these types, and use an integer n to get the required input and assign it to them:

```
enum Choice {ROCK, PAPER, SCISSORS};
enum Winner {PLAYER1, PLAYER2, TIE};
int main()
{ int n;
  Choice choice1, choice2;
  Winner winner;
  cout << "Choose rock (0), paper (1), or scissors (2):" << endl;
  cout << "Player #1: ";
  cin >> n;
  choice1 = Choice(n);

  cout << "Player #2: ";
  cin >> n;
  choice2 = Choice(n);
  if (choice1 == choice2) winner = TIE;
  else if (choice1 == ROCK)
    if (choice2 == PAPER) winner = PLAYER2;
    else winner = PLAYER1;
  else if (choice1 == PAPER)
    if (choice2 == SCISSORS) winner = PLAYER2;
    else winner = PLAYER1;
  else // (choice1 == SCISSORS)
    if (choice2 == ROCK) winner = PLAYER2;
    else winner = PLAYER1;
  if (winner == TIE) cout << "\tYou tied.\n";
  else if (winner == PLAYER1) cout << "\tPlayer #1 wins." << endl;
  else cout << "\tPlayer #2 wins." << endl;
}
```