Department of Computer Science and Engineering
Institute of Technical Education & Research, SOA, Deemed to be University

TAKE HOME ASSIGNMENT-03
Array, Array Processing & Introduction to Pointers
Practical Programming with C (CSE 3544)

Problem Statement:

Experiment with array, it's processing and applications of arrays

Do I Able to Answer? Evaluate:

0. Determine the output of the following code snippet;

```
int main(){
 int x=2,y=5,i;
 int a[]={A(x,y),A(5,5)};
 for(i=0;i<2;i++){
     printf("%d ",a[i]);
 }
 return 0;}
```

```
int A(int x, int y)
{
    return(x*y);
}
```

Output

1. Consider the following C function definition.

```
int f(int x, int y)
{
  for(int i=0;i<y;i++)
  {
    x=x+x+y;
  }
 return x;
}
```

Select the Output **GATE-2024**

Which of the following statements is/are TRUE about the above function?

(A) If the inputs are x=20, y=10, then the return value is greater than 2^{20}

(B) If the inputs are x=20, y=20, then the return value is greater than 2^{20}

(C) If the inputs are x=20, y=10, then the return value is less than 2^{10}

(D) If the inputs are x=10, y=20, then the return value is greater than 2^{20}

2. Consider the following NASI C function; **GATE-2021**

```
int SimpleFunction(int Y[], int n, int x)
{
int total=Y[0], loopIndex;
for(loopIndex=1;loopIndex<=n-1;loopIndex++)
   total=x * total + Y[loopIndex];
return total;
}
```

Trace the Output

Let **Z** be an array of 10 elements with **z[i]=1**, for all i such that $0 \leq i \leq 9$. The value returned by **simpleFunction(z,10,2)** is ____

3. Consider the following C program.

```
#include <stdio.h>
int main() {
 int arr[4][5];
 int i,j;
 for(i=0;i<4;i++){
      for(j=0;j<5;j++)
        arr[i][j]=10*i+j;
 }
 printf("%d %d\n",arr[2][4],arr[1][2]);
 printf("%d %d",arr[3][3], arr[2][3]);
 return 0;
}
```

Trace the Output

4. Determine the output of the following code snippet;

```
int pp(int a, int b)
{
  int arr[20];
  int i, tot=1, ex,len;
  ex=a;
  len=tob(b,arr);
  for(i=0;i<len;i++)
   printf("%d\n",arr[i]);
  for(i=0;i<len;i++){
   if(arr[i]==1)
      tot=tot*ex;
   ex=ex*ex;
  }
  return(tot);
}
```

```
int tob(int b, int *arr){
  int i;
  for(i=0;b>0;i++){
    if(b%2)
        arr[i]=1;
    else
        arr[i]=0;
    b=b/2;
  }
  return(i);
}
int main(){
  printf("%d", pp(3,4));
  return 0;
}
```

Trace the Output

5. Consider the following program:

```
void f1(int a[],
        int n)
{
  a[1]=10;
}
```

```
void f2(int a[],
    int n){
  a[0]=2*n;
  a[4]=n/2;
}
```

```
void f3(int a[],int
     n){
  a[2]=4*n;
  a[3]=50;
}
```

```
int main(){
int a[5],i;
f1(a,5);f2(a,5);
f3(a,5);
for(i=0;i<5;i++)
 printf("%d ",a[i]);
return(0);}
```

Write and describe the output

6. Consider the following C program; **[GATE 2019]**

```
#include<stdio.h>
int main(){
 int arr[]={1,2,3,4,5,6,7,8,9,0,1,2,5};
 int *ip=arr+4;
 printf("%d\n",ip[1]);
 return 0;
}
```

Output and it's reason

The number that will be displayed on execution of the program is ___

7. The value printed by the following program is_____. **[GATE 2016]**

```
#include<stdio.h>
void f(int* p, int m){
 m = m + 5;
 *p = *p + m;
 return;
}
void main(){
int i=5, j=10;
f(&i, j);
printf("%d", i+j);
}
```

Show the desired Output

8. Consider the following C program segment **[GATE 2015]**

```
int main(){
 char s1[7]="1234",*p;
 p=s1+2;
 *p='0';
 printf("%s", s1);
}
```

Draw the array and show the pointer▼

What will be printed by the program
(A) 12 (B) 120400 (C) 1204 (D) 1034

9. The output of the following C program segment is : **GATE2015**

```
int f1(int a, int b){
 int c;
 c=a;
 a=b;
 b=c;
}
```

```
int f2(int *a,int *b){
  int c;
  c=*a;
  *a=*b;
  *b=c;
}
```

```
int main(){
 int a=4,b=5,c=6;
 f1(a,b);
 f2(&b,&c);
 printf("%d",c-a-b);
}
```

Output & it's reason

10. Consider the following program in C language: **GATE-2014**

```
#include <stdio.h>
int main()
{
    int i;
    int *pi=&i;
    scanf("%d",pi);
    printf("%d\n",i+5);
    return 0;
}
```

Select the option ▼

Which one of the following statements is **TRUE**?

(A) Compilation fails.

(B) Execution results in a run-time error.

(C) On execution, the value printed is 5 more than the address of variable i.

(D) On execution, the value printed is 5 more than the integer value entered.

11. Consider the following C function in which **`size`** is the number of elements in the array **`E`**:

```
int MyX(int *E, unsigned int size)
{
 int Y = 0;
 int Z;
 int i, j, k;
 for(i = 0; i < size; i++)
   Y = Y + E[i];
 for(i = 0; i < size; i++)
  for(j = i; j < size; j++)
  {
   Z = 0;
   for(k = i; k <= j; k++)
      Z = Z + E[k];
   if (Z > Y)
     Y = Z;
 }
 return Y;
}
```

Select the option▼ **GATE-2014**

The value returned by the function **`MyX`** is the

(A) maximum possible sum of elements in any sub-array of array **`E`**.

(B) maximum element in any sub-array of array **`E`**.

(C) sum of the maximum elements in all possible sub-arrays of array **`E`**.

(D) the sum of all the elements in the array **`E`**.

Explanation:

12. Consider the C function given below. Assume that the array **`listA`** contains **`n`** (> 0) elements, sorted in ascending order.

```
int ProcessArray(int *listA, int x, int n){
 int i, j, k;
 i=0;
 j=n-1;
 do{
    k=(i+j)/2;
    if(x <= listA[k])
       j = k-1;
    if(listA[k] <= x)
       i = k+1;
   }while(i <= j);
   if (listA[k] == x)
       return(k);
   else
     return -1;
}
```

Select the option▼ **GATE-2014(III)**

Which one of the following statements about the function **`ProcessArray`** is **CORRECT?**

(A) It will run into an infinite loop when **`x`** is not in **`listA`**.

(B) It is an implementation of binary search.

(C) It will always find the maximum element in **`listA`**.

(D) It will return **`-1`** even when **`x`** is present in **`listA`**.

Brief Explanation:

13. Find the output of the following code snippet;

```
#include <stdio.h>
void swap(int *,int *);
int main() {
    int a=10,b=20;
    swap(&a,&b);
    printf("%d %d", a,b);
    return 0;
}
void swap(int *p,int *q){
   p=q;
   *p=100;
   *q=20;
}
```

Draw a digram to show the pointer & write the output▼

14. Find the output of the given code snippet;

```
#define MAX 100
void set(int arr[], int n);
int main() {
    int arr[MAX],n=10,i;
    set(arr,n);
    for(i=0;i<n;i++)
    printf("%d ",arr[i]);
    return 0;
}
void set(int arr[], int n){
   int i;
   for(i=0;i<n;i++)
     if(i%2==0)
        arr[i]=1;
    else
       arr[i]=0;
}
```

Output▼

Write an alternate way for the function prototype

15. Find the output of the code snippet;

```
int main(){
    int i,j;
    int a[8]={1,2,3,4,5,6,7,8};
    for(i=0;i<3;i++){
        a[i]=a[i]+1;
        i++;
    }
    i--;
    for(j=7;j>4;j--){
        int i=j/2;
        a[i]=a[i]-1;
    }
    printf("%d %d\n", i, a[i]);
    for(int i=0;i<8;i++)
      printf("%d ",a[i]);
    return 0;
}
```

Draw the content of the array: Initially & Finally with other output(s)▼

16. Find the output of the code snippet;

```
#define N 50
int fun(int X[], int Y[], int Z[], int n);
int main() {
    int X[N], Y[N], Z[N],n=6,s,i;
    s=fun(X,Y,Z,n);
    for(i=0;i<=n;i++)
      printf("%d ",X[i]);
    printf("\n");
    for(i=0;i<=n;i++)
      printf("%d ",Y[i]);
    printf("\n");
    for(i=0;i<=n;i++)
      printf("%d ",Z[i]);
    printf("\n");
    printf("Last column sum=%d", s);
    return 0;
}
int fun(int X[], int Y[], int Z[], int n)
{
    int i;
    X[0]=Y[0]=Z[0]=0;
    X[1]=1;Y[1]=2;Z[1]=3;
    for(i=2;i<=n;i++){
        X[i]=Y[i-1]+Z[i-2];
        Y[i]=2*X[i];
        Z[i]=3*Y[i];
    }
    return(Y[n]+Z[n]+X[n]);
}
```

Output ▼

17. Find the output of the code snippet;

```
int main(){
   char p[20],str[]="STRING";
   int l=0;
   for(l=0;str[l]!='\0';l++);
   l=l-1;
   for(int i=0;i<=l;i++){
       p[i]=str[l-i];
   }
   printf("%s", p);
   return 0;}
```

Briefly Describe the Output▼

18. Find the output of the code snippet;

```
int main() {
   char p[20];
   char str[]="STRING";
   int l=0;
   for(l=0;str[l]!='\0';l++);
   for(int i=0;i<=l;i++){
       p[i]=str[l-i];
   }
   printf("%s", p);
   return 0;}
```

Select the option with the content of the array `p` ▼

The output of the program is

(A) GNIRTS

(B) STRING

(C) GNIRT

(D) No output is printed

19. Find the output of the code snippet;

```
void cse(int *, int);
int main() {
   int a=111;
   printf("%d\n",a);
   cse(&a,a);
   printf("%d\n",a);
   printf("%d\n",*(&a));
   return 0;
}
void cse(int *x, int y)
{
    *x=222;
}
```

Briefly Describe the output ▼

20. Find the output of the code snippet;

```
int main(){
 int rows=3, cols=4;
 int a[3][4]={1,2,3,4,5,6,7,8,9,10,11,12};
 int i=80,j=90,k=99;
 for(i=0;i<rows;i++){
  for(j=0;j<cols;j++)
    printf("%d ",a[i][j]);
  printf("\n");
 }
 for(i=0;i<rows;i++)
  for(j=0;j<cols;j++)
    if(a[i][j]<k)
       k=a[i][j];
 printf("%d",k);
 return 0;
}
```

Output▼

21. Predict the output of the given code snippet:

```
int main(){
 char i;
 for(i='A';i<='Z';++i)
   printf("%d ",i);
 return 0;
}
```

Output▼

22. Consider the following function implemented in C: **GATE-2017, II**

```
void printxy (int x, int y)
{
 int *ptr ;
 x = 0;
 ptr = &x;
 y = * ptr;
 * ptr = 1;
 print f ("%d, %d" x, y);
}
```

Select & Trace Output▼

(A) 1, 0 (B) 1, 1 (C) 0, 1 (D) 0, 0

The output of invoking **`printxy(1,1)`** is

23. Consider the following snippet of a C program. Assume that swap (&x, &y) exchanges the contents of x and y. **GATE-2017, II**

```
int main ( ) {
 int array[]={3,5,1,4,6,2};
 int done =0 ;
 int i ;
 while (done = = 0) {
    done = 1;
    for (i = 0; i <=4; i ++) {
     if(array [i] < array [i +1]) {
       swap(&array [i], &array [i+1]);
       done = 0;
    }
 }
 for (i = 5 ; i > =1; i --) {
  if(array [i] > array [ i-1]) {
     swap (&array [i] , &array [i-1]);
     done = 0;
  }
 }
}
printf ( " %d " , array [3] );
}
```

Select the option & Trace the Output▼

The output of the program is....
(A) 5 (B) 4 (C) 3 (D) 2

24. Consider the following C program. **GATE-2017**

```
#include<stdio.h>
#include<string.h>
void printlength (char *s, char *t) {
 unsigned int c = 0;
 int len = ((strlen(s) - strlen (t)) > c) ?
    strlen(s): strlen(t);
 printf ("%d\n", len);
}
void main(){
 char *x="abc";
 char *y="defgh";
 printlength(x,y);
}
```

Recall that **strlen** is defined in **string.h** as returning a value of type **size_t**, which is an unsigned int. The output of the program is .....

Select the option and Trace the Output▼

(A) 5 (B) 3 (C) 4 (D) 6