Structured Programming Language

Assignmen - 02

MD MUSFIQUR RAHMAN

011 221 334

Section : F

Submitted to

Anika Tusmim Rodela

Set - C

① ⓐ

```
#include<stdio.h>
#include<math.h>
int main() {
    float n, m=5;
    scanf("%f", &n);
    float P = (n%m)/sqrt(6);
    printf("%f", P);
    return 0;
}
```

ⓑ · 1time – invalid · for numanical value at first.

$VALUE – invalid for dollen sign at finst.

"nyvalue" – invalid for rotation symbol.

last-name – invalid for · hyphen

② int a = 10.0/3*10;

= 33

int e = 0

float b = 1.000000

float d = 3.000000.

② ⓐ

for b=4	for b=5	for b=10	for b=12
Begin	Begin	Begin	Begin
CSE	UIU	UIU	UIU
End	End	End	End

(b)

```
#include<stdio.h>
int main()
{
char catagory;
int year_expenience;
int familyMember;
float family_Pm_Income;   scanF("%c %d %d %d", &catagory, &year_expeniene
                                 &familyMember, &family_Pm_I
if(( year_expenience >= 12 && familyMember > 5) || famil_Pm_Iincome < 1000750)
    { printf("Will recieve bonus");
    }
else if ((catagory == 'Y' || catagory == 'Z') && familyMember > 8 && family_
        pm_Income < 1100.78)
    { printf ("Will recieve bonus");
    }
elseif ( catagory == 'X' && familyMember > 5)
    { printf ("Will receive bonus");
    }
else {
    printf ("Sorry, not eligible for the bonus");
    }
return 0;
}
```

③ⓐ int n=5 , sum = 0 ; i , a = 3 , sign = 1

i = 1

1 <= 5

sum = 3

a = 7

sign = -1

i = 2

2 <= 5

sum = 3 + 7 - 1 = -4

a = 13

sign = 1

i = 3

3 <= 5

sum = -4 + 13 * 1 = 9

a = 17

sign = -1

i = 4

4 <= 5

sum = 9 + 17 (-1) = -8

a = 23

sign = 1

i = 5

5 <= 5

sum = -8 + 23 * 1 = 15

a = 27

sign = -1

i = 6

out from the loop

finally

sum = 15 .

⑥

```
#include<stdio.h>
#include<math.h>
int divSum (int n)
{int i, sum = 0;
for(int i = 1; i <= (sqrt(n)); i++) {
  if ( n%i == 0) {
    if( n/i == i ) {
        sum = sum + i;
    }
    else {
        sum = sum + i;
        sum = sum + (n/i)
    }
  }
  return sum;
}
int isDef (int n)
  {
   return (divSum(n) < (2*n));
  }
void main ( )
{
  char A[100];
   int n;
  scanf ("%d", &n);
  if ( isDef (n) == 0 ) {
    printf (" NOT deficient ");
  }
  else {
    printf (" Yes deficient ");
  }
```

4(a)

```
int F[6]={0};
int i;
F[0] = 1;
F[1] = 1
for(i=2;i<=5;i++){
  F[i] = F[i-1]+F[i-2];
 printf("%d %d %d\n", F[i-2], F[i-1], F[i]);
}
printf("%d %d %d ", F[i-2], F[i-1], F[i-1]+F[i-2]);
```

i	F[0]	F[1]	F[i]	F[i-2]	F[i-1]	F[2]	F[3]	F[4]	F[5]
2	1	1	2	1	1	2	-	-	-
3	1	1	3	1	2	2	3	-	-
4	1	1	5	2	3	2	3	5	-
5	1	1	8	3	5	2	3	5	8
6	False	F	F	F	F	F	F	F	F

For i=6 the loop is End the

print

F[i-2] = F[6-2] = F[4] = 5

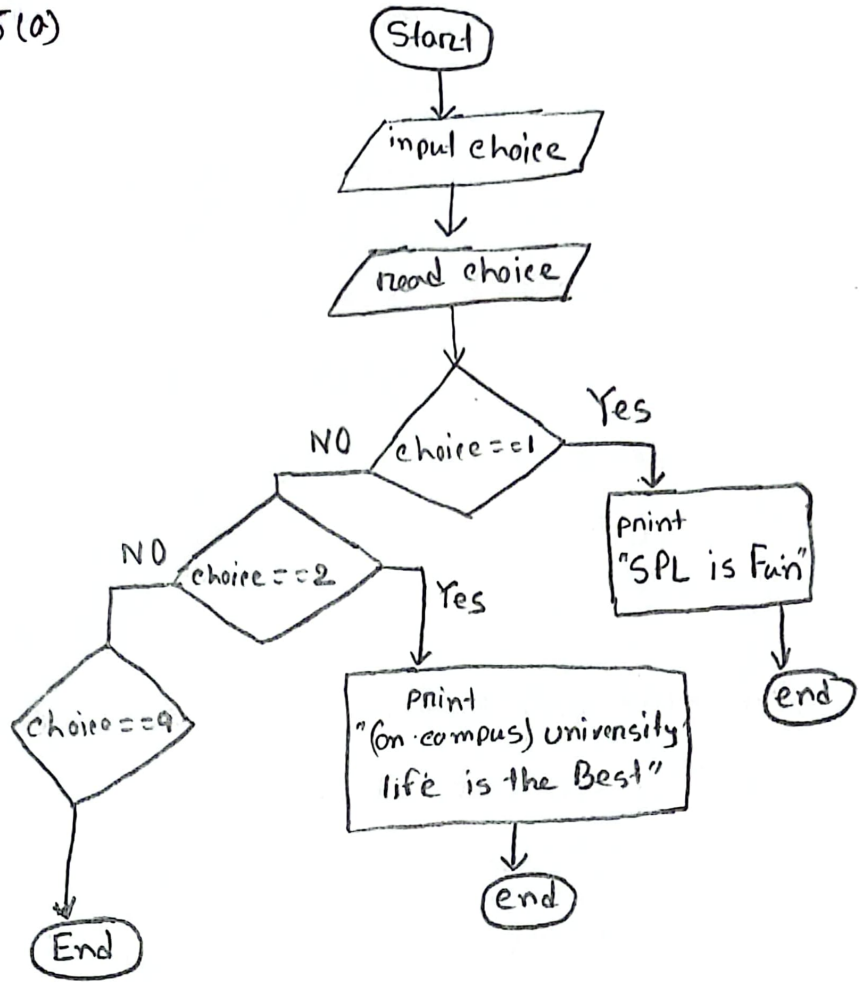
F[i-1] = P[6-1] = F[5] = 8

F[5]+F[4] = 8+5 = 13.

4(b)

```
#include<stdio.h>
int main(){
int i,n, max, max_index, min, min_index;
scanf("%d", &n);
int array[n];
for(i=0; i<n; i++){
    scanf("%d", &array[i]);
}
max = array[0];
min = array[0];
max_index = 0;
min_index = 0;
for(i=1; i<n; i++) {
    if(array[i]>max)
    {max = array[i];
     max_index = i;
    }
    if(array[i]<min){
    {min = array[i];
     min_index = i;
    }
}
printf("Max : %d Index : %d", max, max_index);
printf("Min : %d Index : %d", min, min_index);
return 0;
}
```

5(a)



(b)

```
#include<stdio.h>
int main(){
int width, i, j;
  scanf("%d", &width);

for(i=1; i<=width; i++){
    for(j=1; j<=width; j++){
        if(j==i || j==width-i+1){
            printf("x");
        }else{
         printf(" ");
        }
        printf("\n");
    }
return 0;
}
```

Set-D

Q1

(a)
```
#include<stdio.h>
void main( ) {
  int num1= 5; float num2; char chr = '9';
scanf ("%f ", &num2);
 num1 = (int) num2 % chr;
 printf ("Result is = %d ", num1);
}
```

(b) smallest-val → invalid for hyphen(-).

while → invalid for this is loop

2nd Num → invalid for numerical value at first

!New → invalid for condition check statment

avg mark → invalid for space

(c) float a = 10.000000

int b = 10

float c = 12.500000

int d = 12

Q2.

(a) for num=1

12 5 12

for num = 3

0 0 0

(b) int sum = 0, i, a = 1, b, x = 1, y = 1.

i = 1

1 <= 5 → T

sum = 0 + 1 = 1

b = 6 * 1 + 1 = 7

a = 1 + 7 = 8

y = 2

x = 1 + 2 = 3

i = 2

2 <= 5 → T

sum = 1 + 8 = 9

b = 6 * 3 - 11 = 19

a = 8 + 19 = 27

y = 3

x = 6

i = 3

3 <= 5 → T

sum = 9 + 27 = 36

b = 6 * 6 + 1 = 37

a = 27 + 37 = 64

y = 4

x = 6 + 4 = 10

i = 4

4 <= 5

sum = 36 + 64 = 100

b = 6 * 10 + 1 = 61

a = 64 + 61 = 125

y = 5

x = 10 + 5 = 15

i = 5

5 <= 5

sum = 100 + 125 = 225

b = 6 * 15 + 1 = 91

a = 125 + 91 = 216

y = 6

x = 15 + 6 = 21

i = 6

6 <= 5

condition false

end of the loop

finally

i = 6, a = 216, b = 91, x = 21, y = 6.

Q3

@ void main()

```
int n = 3, i, j, sum = 0;

do {
    j=0;
    do {
        i=0;
        if(i==j)
            sum += i+j;
        elseif(i>j)
            sum += i+n;
        else
            sum += n-j;
        j++;
    } while(j<=i);
    i++;
} while(i<n);
}
```

(b)
```
#include<stdio.h>
int main() {
    int count =0;
    double sum=0;0;
    double inp;
while(1){
    scanf("%df", &inp);
if(inp==0) {
    break;
}else if (inp>0) {
    sum+= input;
    count ++
}
else
    { printf("Enter positive number:\n");
}
};if (count>0) {
    double avg = sum/count;
    printf("%.2lf\n", count, avg);
}else {
    printf("sorry");
}
return 0;
}
```

4a

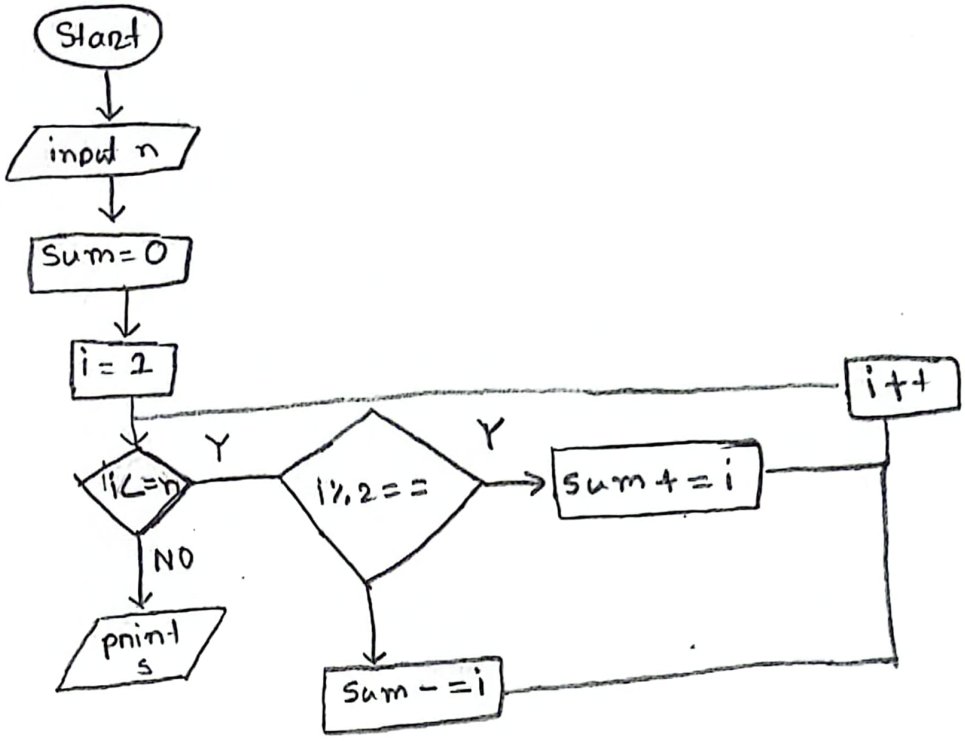
```
F[i] = n+i
if (F[i]%2 == 0) {
    F[i] *= 2
}
```

n	i	F[0]	F[1]	F[2]	F[3]	F[4]	F[5]
garb.	garb.	0	0	0	0	0	0
3	0	3	0	0	0	0	0
3	1	3	4	0	0	0	0
3	1	3	8	0	0	0	0
3	2	3	8	5	0	0	0
3	3	3	8	5	6	0	0
3	3	3	8	5	12	0	0
3	4	3	8	5	12	7	0
3	5	3	8	5	12	7	8
3	5	3	8	5	12	7	16
3	6	—	—	—	—	—	—

for i = 6 the loop is end because 6 <= 5 the condition is false, so the loop is end.

4(b)
```
#include<stdio.h>
int main() {
 int n, i;
 scanf("%d", &n);
 int arr[100] = { };
 for(i=0; i<n; i++) {
    scanf("%d", &arr[i]);
  }
printf("Index \t value \t");
printf("---- \t ---- \n");
for(i=0; i<n; i++) {
 int sum=0;
 int count=0;
for(int i=0; i<n; i++){
  if(i%2!=0) {
       sumt = arr[i];
       count ++;
  }
}
 if(count>0) {
      double avg = (double) sum/count );
      printf("%.5lf \n", avg);
  } else
     printf("NO - odd- indexed element to calculate the avg");
 }
 return 0;
}
```

5a



5b

```
#include<stdio.h>
int main( ) {
   int n;
   scanf ("%d ", &n);

  int result = 2*n;
 for (int i = n; i>=1 ; i--) {

    for ( int j = result ; j>=2; j-= 2 ){

        printf ("%d", j);

     }
     printf ("\n");
     result -= 2;

 }
 return 0;

}
```