CS 101 Revision Session

Autumn 2024-25

Durgam Latha and Abhijat Bharadwaj

Question 1

Find the output:

```
int a[5] = {1,2,3,4,5};

int sum = 0;

for (int i = 0; i<5; i++){
    sum += a[i];
}

cout << sum << endl;
```

Solution 1

```
int a[5] = {1,2,3,4,5};

int sum = 0;

for (int i = 0; i<5; i++){
    sum += a[i];
}

cout << sum << endl;
```

Output: 15

- Output the sum of values in array a

Question 2

Find the problem in the code and fix it

Goal of code - Store the first 10 multiples of 5 in a

```
int a[10] = {};

for (int i=1; i<11; i++){
      a[i] = 5*i;
}
```

Solution 2

Goal of code - Store the first 10 multiples of 5

```
int a[10] = {};

for (int i=1; i<11; i++){
      a[i] = 5*i;
}
```

Error: i=10 is out of index

Fix:

- index range is [0,9]
- int i=0;i<10;i++

- and also we need to update the value assignment to a
- a[i] = 5*(i+1)

Solution 2

Corrected Code

```
int a[10] = {};

for (int i=0; i<10; i++){
        a[i] = 5*(i+1);
}
```

Question 3

Find output:

```
int n = 5;
int X[n] = {10,15,20,25,30};

cout << X[5] << endl;
```

Solution 3

```
int n = 5;
int X[n] = {10,15,20,25,30};

cout << X[5] << endl;
```

Output: Some random value or can crash because the index is out of range

Question 4

```
typedef int int05 [5];

int05 Y = {3,6,9,12,15};
int05& Z = Y;

Z[3] = Z[3] * Y[3];

for (int i=0; i<5; i++) {
        cout << Y[i] << “ “;
}
```

Find the output

Solution 4

```
typedef int int05 [5];

int05 Y = {3,6,9,12,15};
int05& Z = Y;

Z[3] = Z[3] * Y[3];

for (int i=0; i<5; i++) {
        cout << Y[i] << “ “;
}
```

Output: 3 6 9 144 15

Explanation:
Z is a reference to Y so changes in Z also reflects in Y

Question 5

Find Output

```
void printarray (int A[], int n)
{
        for (int i=0; i<n; i++) {
                cout << A[i]  << “ ”;
        }
}
void inc_array (int A[], int n)
{
        for (int i=0; i<n; i++) {
                A[i] = A[i] +1;
        }
}
```

```
int main()
{

        int X[10] = {1,2,3,4,5,6,7,8,9,10};

        inc_array(X,5);

        printarray(X,10);

}
```

Solution 5

Output - 2 3 4 5 6 6 7 8 9 10

Explanation - Arrays are passed by reference in functions.

inc_array(X,5) - Only first 5 elements of X are increased by 1

Question 6

Find the errors and fix it

```
typedef int int_10 [10];

int_10 max(int_10& A, int_10& B){
        for (int i=0; i<10; i++){
                A[i] = (A[i]>B[i])? A[i]:B[i];
        }
        return A;
}
```

```
int main() {
        int_10 X = {1,2,3,4,5,6,7,8,9,0};
        int_10 Y = X;

        int_10 Z = max(X,Y);

        for (int i=0; i<10; i++) {
                cout << Z[i] << “ ”;
        }
}
```

Solution 6

```
typedef int int_10 [10];

int_10 max(int_10& A, int_10& B){
        for (int i=0; i<10; i++){
                A[i] = (A[i]>B[i])? A[i]:B[i];
        }
        return A;
}
```

Error:

1) int_10 max(...) {...}
   - Function cannot return array

Fix:

```
int_10& max(int_10& A, int_10&B)
```

- Function can return reference to Array

Solution 6

```
int main() {
        int_10 X = {1,2,3,4,5,6,7,8,9,0};
        int_10 Y = X;

        int_10 Z = max(X,Y);

        for (int i=0; i<10; i++) {
                cout << Z[i] << “ ”;
        }
}
```

Error:

2) int_10 Y= X
   - Array can be initialized only using braces {}

Fix:

Initialize Y with some values

- for example:
  int_10 Y = {2,3,4,5,2,4,8,9,4,1};

Solution 6

```
int main() {
    int_10 X = {1,2,3,4,5,6,7,8,9,0};
    int_10 Y = X;

    int_10 Z = max(X,Y);

    for (int i=0; i<10; i++) {
        cout << Z[i] << “ ”;
    }
}
```

Error:

3) `int_10 Z= max(X,Y);`
   - Array can be initialized only using braces {} and max function cannot return array. So, in any case there is error

Fix:

```
int_10& Z = max(X,Y);
```

Solution 6

Corrected Code

```
typedef int int_10 [10];

int_10& max(int_10& A, int_10& B){
    for (int i=0; i<10; i++){
        A[i] = (A[i]>B[i])? A[i]:B[i];
    }
    return A;
}
```

```
int main() {
    int_10 X = {1,2,3,4,5,6,7,8,9,0};
    int_10 Y = {2,3,4,5,2,4,8,9,4,1};

    int_10& Z = max(X,Y);

    for (int i=0; i<10; i++) {
        cout << Z[i] << “ ”;
    }
}
```

Question 7

What is output of modified code of Question 6?

```
typedef int int_10 [10];

int_10& max(int_10& A, int_10& B){
    for (int i=0; i<10; i++){
            A[i] = (A[i]>B[i])? A[i]:B[i];
    }
    return A;
}
```

```
int main() {
    int_10 X = {1,2,3,4,5,6,7,8,9,0};
    int_10 Y = {2,3,4,5,2,4,8,9,4,1};

    int_10& Z = max(X,Y);

    for (int i=0; i<10; i++) {
        cout << Z[i] << “ ”;
    }
}
```

Solution 7

Output - 2 3 4 5 5 6 8 9 9 1

Explanation:

- Function max returns the maximum of A[i] and B[i] elements at same index and stores in A[i].
- Z references to A in max function. So Z stores the maximum value of X[i]and Y[i]

Question 8

Fill the Blanks

Goal - Output the reverse sequence of input

Hint - Use stack concept

Input:	Output:
a	f
b	e
c	d
d	c
e	b
f	a

```
int Nmax = 6;

char A[Nmax];
int top=-1;
char a;
cin >> a;
while(top<(Nmax-1) && a>='a' &&
a<='z')
{
        A[______] = a; cin >> a;
}

while(_______)
{
        cout << A[______]<<endl;
}
```

Solution 8

1) Top points to the index of last entry so before storing new entry we have to increment the top value.

   Increase top value by 1 before using it.

   A[++top] = a;

```
int Nmax = 6;

char A[Nmax];
int top=-1;
char a;
cin >> a;
while(top<(Nmax-1) && a>='a' &&
a<='z')
{
        A[++top] = a; cin >> a;
}

while(______)
{
        cout << A[____]<<endl;
}
```

Solution 8

2) Print the output till the top value is Zero. So

```
while(top>=0)
```

```
int Nmax = 6;

char A[Nmax];
int top=-1;
char a;
cin >> a;
while(top<(Nmax-1) && a>='a' &&
a<='z')
{
        A[++top] = a; cin >> a;
}

while(top>=0)
{
        cout << A[____]<<endl;
}
```

Solution 8

3) We have to decrease the top value after using the value of top.

So Decrease the value of top after using the value - post decrement

```
cout << A[top--] << endl;
```

Final Code ====>

```
int Nmax = 6;

char A[Nmax];
int top=-1;
char a;
cin >> a;
while(top<(Nmax-1) && a>='a' &&
a<='z')
{
        A[++top] = a; cin >> a;
}

while(top>=0)
{
        cout << A[top--]<<endl;
}
```

Question 9

Fill in the blanks

Goal - Find factorial of n

Hint - Use concept of recursive functions

```
int factorial (int n) {
        if (________) {return _______;}
        return (_________);
}

int main() {
        int n = 5;

        int n_factorial = __________;

        cout << n_factorial ;
}
```

Solution 9

Factorial of n = n! = 1*2*3*4*.....*n

1) Base Cases:

   0! = 1

   1! = 1

```
int factorial (int n) {
        if (n==0 || n==1) {return 1;}
        return (__________);
}

int main() {
        int n = 5;

        int n_factorial = __________;

        cout << n_factorial ;
}
```

Solution 9

Factorial of n = n! = 1*2*3*4*.....*n

2) Recursing Condition:

n! = (n-1)! * n

```
int factorial(int n) {
        if (n==0 || n==1) {return 1;}
        return ( n * factorial(n-1));
}

int main() {
        int n = 5;

        int n_factorial = __________;

        cout << n_factorial ;
}
```

Solution 9

Factorial of n = n! = 1*2*3*4*.....*n

3) Call the function

factorial(n)

Final Code ======>

```
int factorial(int n) {
        if (n==0 || n==1) {return 1;}
        return ( n * factorial(n-1));
}

int main() {
        int n = 5;

        int n_factorial = factorial(n);

        cout << n_factorial ;
}
```

Question 10

```
int func(int arr[], int n) {
   if (n == 0) return 0;
   return arr[--n] + func(arr, n--);
}
int main() {
    int arr[] = {1, 2, 3, 4, 5};
    int n = 5;
    cout<< "Value :" << func(arr, n) << endl;
    return 0;
}
```

Find Output

Question 10

```
int func(int arr[], int n) {
    if (n == 0) return 0;
    return arr[--n] + func(arr, n--);
}
int main() {
     int arr[] = {1, 2, 3, 4, 5};
     int n = 5;
     cout<< "Value :" << func(arr, n) << endl;
     return 0;
}
```

Do we need this *n*--?

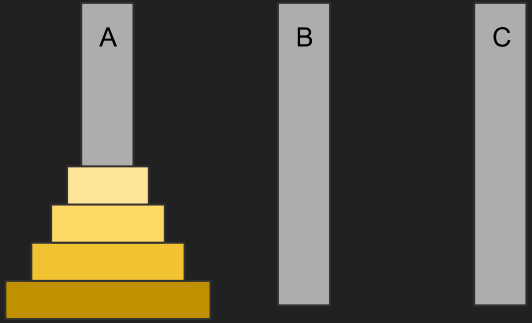
Answer: 15

Question 11: Towers of Hanoi

You are given 3 rods, named ‘A’, ‘B’ and ‘C’ and *n* different-sized annular-discs. Initially, all discs are stacked in decreasing order of their size on rod ‘A’. You are tasked to move them to another rod ‘B’, obeying the following rules:

- Only one disc can be moved at a time.
- A disc can only be moved if it is the uppermost disc on a stack.
- A moved disc can only be placed on the top of another stack
- No disc can be placed on top of a smaller disc.

Complete the code to help you out!



Question 11: Towers of Hanoi

```
void towerOfHanoi(int n, char from, char to, char aux){
    if (n == 0) {
        return;
    }
    towerOfHanoi(n - 1, -----, -----, -----);
    cout << "Move disc " << n << " from rod " << from
         << " to rod " << to << endl;
    towerOfHanoi(n - 1, -----, -----, -----);
}
```

Solution 11: Towers of Hanoi

```
void towerOfHanoi(int n, char from, char to, char aux){
    if (n == 0) {
        return;
    }
    towerOfHanoi(n - 1, from, aux, to);
    cout << "Move disc " << n << " from rod " << from
         << " to rod " << to << endl;
    towerOfHanoi(n - 1, aux, to, from);
}
```

Question 12: Telescopic Sum

Fill in The Blanks

We are trying to find the sum $\sum_{n=1}^{N} \frac{1}{n(n+1)}$

```
double telescopicSum(int N) {
    if (___(i)___) return 1.0 / (1 * 2);
    return 1.0 / (N * (N + 1)) + ___(ii)___;
}
```

Question 12: Telescopic Sum

Comment: Do you think recursion is always the best solution?

Solution

```
double telescopicSum(int N) {
    if (N == 1) return 1.0 / (1 * 2);
    return 1.0 / (N * (N + 1)) + telescopicSum(N - 1);
}
```

Question 13: Check for Powers

You are given an integer. Complete the code to find whether it is a power of k.

```
bool check(int n, int k) {
    __________
    __________
    __________
}
```

Question 13: Check for Powers

You are given an integer. Complete the code to find whether it is a power of k.

```
bool check(int n, int k) {
    if(n<=0)return false;
    __________
    __________
}
```

Question 13: Check for Powers

You are given an integer. Complete the code to find whether it is a power of k.

```
bool check(int n, int k) {
    if(n<=0)return false;
    if(n==1)return true;
    _______
}
```

Question 13: Check for Powers

You are given an integer. Complete the code to find whether it is a power of k.

```
bool check(int n, int k) {
    if(n<=0)return false;
    if(n==1)return true;
    return (n%k ==0 && check(n/k,k));
}
```

Question 14: Max of Array

Find the problem in the code. Aim is to find the maximum of the given array.

```
int findMax(int arr[], int n) {
    if (n == 1) return arr[0];
    int maxRest = findMax(&arr[1], n--);
    return (arr[0] > maxRest) ? arr[0] : maxRest;
}
int main() {
    int arr[] = {1, 4, 3, -5, 9, 2}; int n = 6;
    cout << findMax(arr, n);
    return 0;
}
```

Question 14: Max of Array

Find the problem in the code. Aim is to find the maximum of the given array.

```
int findMax(int arr[], int n) {
    if (n == 1) return arr[0];
    int maxRest = findMax(&arr[1], n--);
    return (arr[0] > maxRest) ? arr[0] : maxRest;
}
int main() {
    int arr[] = {1, 4, 3, -5, 9, 2}; int n = 6;
    cout << findMax(arr, n);
    return 0;
}
```

Question 14: Max of Array

Find the problem in the code. Aim is to find the maximum of the given array.

```
int findMax(int arr[], int n) {
    if (n == 1) return arr[0];
    int maxRest = findMax(&arr[1], --n);
    return (arr[0] > maxRest) ? arr[0] : maxRest;
}
int main() {
    int arr[] = {1, 4, 3, -5, 9, 2}; int n = 6;
    cout << findMax(arr, n);
    return 0;
}
```

Why &arr[1]?
Think about it!

THANK YOU!