

Unit 3 Data Structure

Stack and Queues

Last five year CBSE Exams Question Answer With Solution

- 1) Write a function in C++ to insert in a static circular Queue containing Players information (represented with the help of array of structure PLAYER). [CBSE 2012]
Assume the following definition of MYNODE for the same:

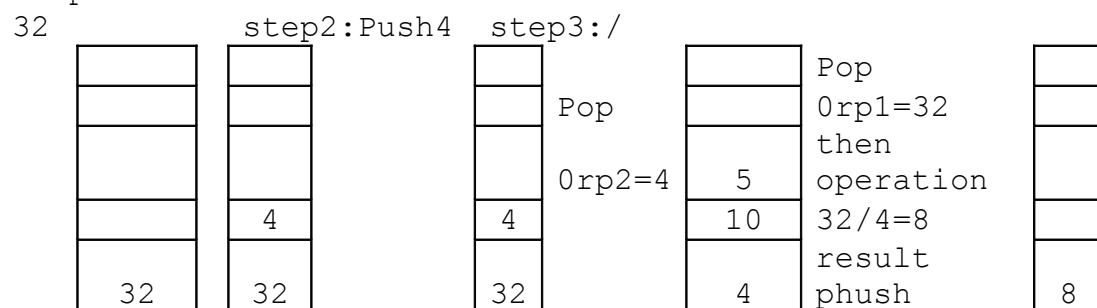
```
struct PLAYER
{
    int PID;
    char Pname[20]
    NODE *Next
};
```

Solution:

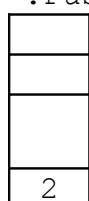
```
void QUESINSERT(PLAYER P[ ], int Front, int &Rear, int N)
{
    if (Front-1=Rear) || (Front==0 && Rear=N-1)
        cout<<"Overflow!! Queue full"<<end;
    else
    {
        Rear=(Rear+1)%N;
        cout<<"Enter Player ID"
        cin>>P[Rear].PID;
        cout<<"Enter Player name";
        gets(P[Rear].Pname;
    }
}
```

- 2) Evaluate the following postfix notation of expression (Show status of stack after execution of each operation): **32, 4, /, 2, *, 12, 3, -, +** [CBSE 2012]

Solution: scanning given postfix expression from left to right
Step1: Push



step 4
: Push 2



step5: *

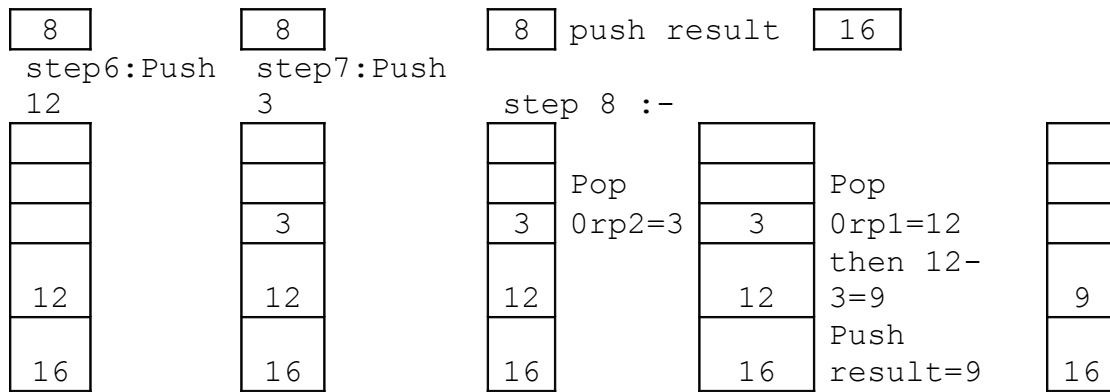


Pop
Op2=2



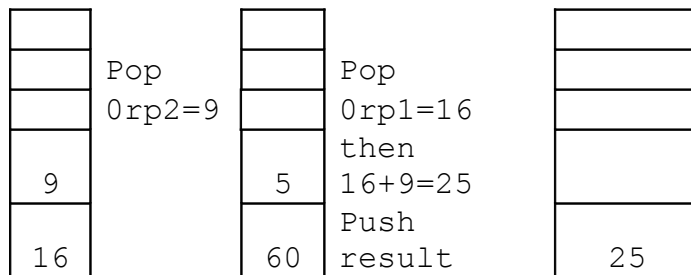
Pop
Op1=8
then
operation
8*2=8





step9 :

+



Pop the
Result=25

- 3) Write a function in C++ to perform Push operation on dynamically allocated stack containing real number. [CBSE 2012]

SOLUTION:

```

struct NODE
{
    Float data; NODE *link;
}
class stack
{
    Node *top;
public:
    stack()
    void Push()
    void Pop();
    void display();
    ~stack();
};
void stack::Push()
{
    NODE *temp;
    temp=new Node;
    cin>>temp->data;
    temp->link=top;
    top=temp;
}

```

- 4) Write a function in QUEINS() in C++ to delete a dynamically allocated Queue containing nodes of following given structure:
[Delhi 2009]

Assume the following definition of MYNODE for the same:

```
struct NODE
{
    int PID;
    char Pname[20]
    NODE *Next
};
```

Solution: The function is:

```
Node *QUEINS(Node *rear, int cal, char val[])
{
    NODE *temp;
    temp =new Node;
    temp->PID=val;
    strcpy(temp->Pname, val);
    temp-> Next=NULL
    rear->Next=temp;
    rear=temp;
    return(rear);
}
```

- 5) Write a function in C++ to delete a dynamically allocated Queue where each node contains a reap number as data [AI 2008]

Assume the following definition of MYNODE for the same:

```
struct MYNODE
{
    float NUM;
    MYNODE *Link;
};
```

Solution: The function is:

```
Void QUEUE::DELETE ()
{
    MYNODE *temp;
    if (front==Null)
    {
        cout<<"Queue Empty";
        val=-1;
    }
    else
    {
        temp=front;
        front=front->link;
        val=temp->data;
        temp->link=null;
        Delete temp;
    }
}
```

```

    }
}

```

- 6) Write a function in C++ to **insert** an element into a **dynamically allocated Queue** where each node contains a name (of type string) as data. **[Delhi 2008]**

Assume the following definition of THENODE for the same.

```

struct THENODE
{
char Name[20];
THENODE *Link;
};

```

Solution:

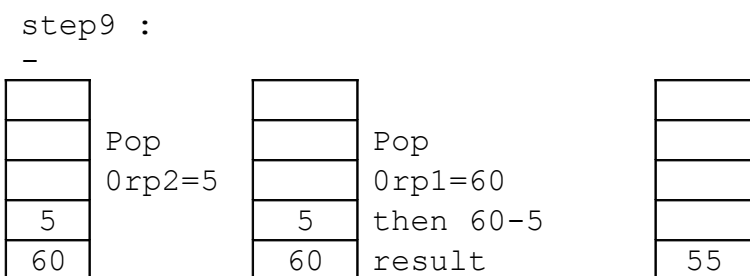
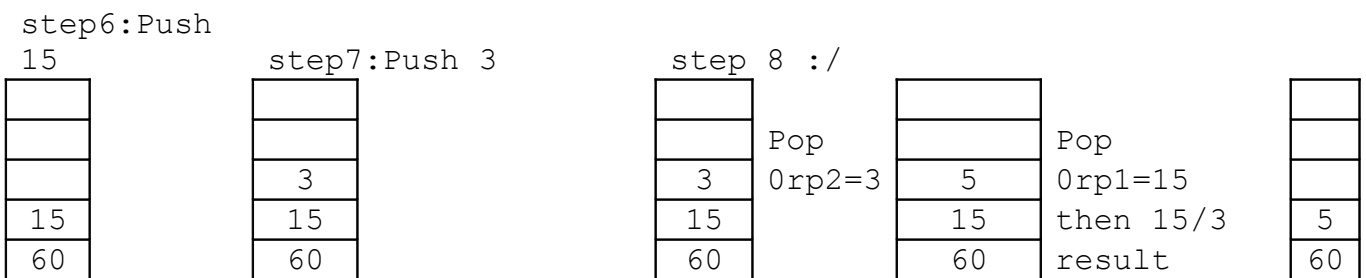
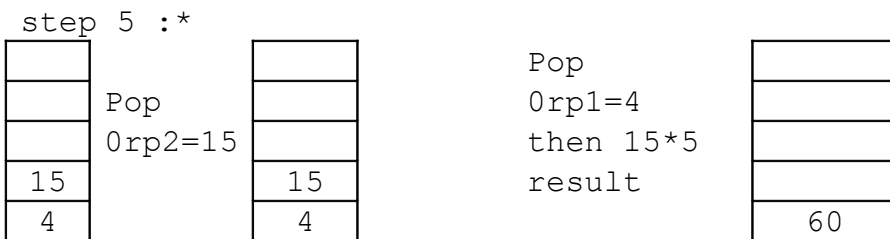
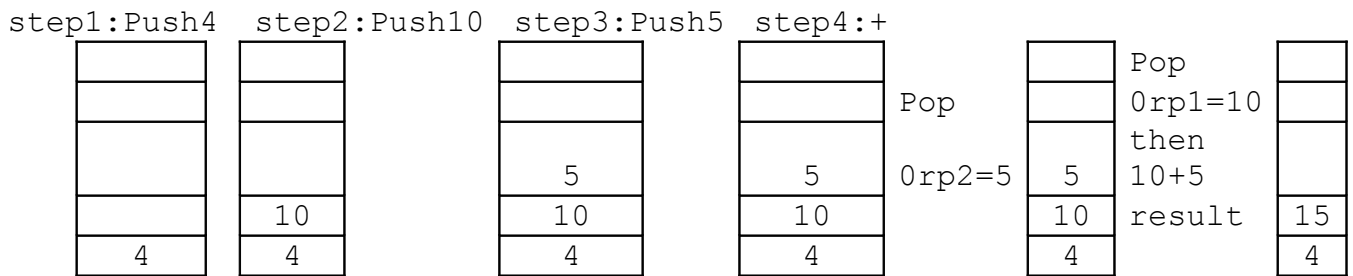
```

struct THENODE
{
    char Name[20];
    THENODE *Link;
};
class Queue
{
THENODE *front,*rear;
public:Queue( )
{
front = rear = NULL;
}
void Insert( );
void Delete( );
void Display( );
};
void Queue::Insert( )
{
THENODE *ptr;
ptr=new THENODE;
if(ptr== NULL)
{
cout<<"\nNo memory to create a new node";
exit(1);
}
cout<<"\nEnter the name";
gets(ptr->Name);
ptr->Link=NULL;
if(rear== NULL)
front=rear=ptr;
else
{
rear->Link=ptr;
rear=ptr;
}
}
}

```

7) Evaluate the following postfix notation of expression (Show status of stack after execution of each operation): **4, 10, 5, +, *, 15, 3, /, -** 2

Solution :



8) Write a function in C++ to **Delete** an element into a **dynamically allocated Queue** where each node contains a real number as data. **[OUTSIDE DELHI 2008:]**

Assume the following definition of MYNODE for the same:

```
struct MYNODE
{
```

```
float NUM;
MYNODE * Link;
};
```

Solution:

```
struct MYNODE
{
float NUM;
MYNODE *Link;
};
class Queue
{
MYNODE *front,*rear;
public:Queue( )
{ front=rear=NULL; }
void Insert( );
void Delete( );
void Display( );
};
void Queue::Delete( )
{
MYNODE *temp;
if(front== NULL)
cout<<"Queue Underflow";
else
{
cout<<"\nThe content of the element to
delete:"<<front->NUM;
temp=front;
front=front->Link;
delete temp;
}
}
```

- 9) **Evaluate** the following **postfix** notation of expression (Show status of stack after execution of each operations): 5, 20, 15, -, *, 25, 2, *, +

Solution:

step	Input symbols	Action	Stack status	Intermediate output
1	5	Push	5	
2	20	Push	5 20	
3	15	Push	5 20 15	
4	-	Pop twice(Binary operator)	5	20-15=5
5	5	Push	5 5	
6	*	Pop twice(Binary operator)		5*5=25
7	25	Push	25	
8	25	Push	25 25	

9	2	Push	25 25 2	
10	*	Pop twice(Binary operator)	25	25*2=50
11	50	Push	25 50	
12	+	Pop twice(Binary operator)		25+50=75
	75	Push	75	
		No More Element		75 result

- 10) Write a function in C++ to **delete** a node containing Book.s information, from a **dynamically allocated Stack** of Books implemented with the help of the following **DELHI : 2007** structure.

```
struct Book
{ int BNo ;
char BName[20] ;
Book *Next ;
} ;
```

Solution:

```
struct Book
{ int BNo ;
char BName[20] ;
Book *Next ;
} ;
class Stack
{ Book *Top;
public:Stack( )
{ Top = NULL; }
void Push( );
void Pop( );
void Display( );
};
void Stack::Pop( )
{ Book *Temp;
If( Top= = NULL)
cout<<"Stack Underflow";
else
{ cout<<"\nThe Book number of the element to
delete:"<<Top->BNo;
cout<<"\nThe Book name of the element to delete:"<<Top-
>BName;
Temp=Top;
Top=Top->Next;
Delete Temp;
}
}
```

- 11) **Evaluate** the following **postfix** notation of expression : 25 8 3
- / 6 * 10+

Solution:

step	Input symbols	Action	Stack status	Intermediate output
1	25	Push	25	
2	8	Push	25 8	
3	3	Push	25 8 3	
4	-	Pop twice(Binary operator)	25	8-3=5
5	5	Push	25 5	
6	/	Pop twice(Binary operator)		25/5=5
7	5	Push	5	
8	6	Push	5 6	
9	*	Pop twice(Binary operator)		5*6=30
10	30	Push	30	
11	10	Push	30 10	
12	+	Pop twice(Binary operator)		30+10=40
	40	Push	40	
		No More Element		40 result

- 12) Write a function in C++ to **delete** a node containing customer.s information, from a **dynamically allocated Queue** of Customers implemented with the help of the following structure:

```
struct Customer
{
    int CNo ;
    char CName[20] ;
    Customer *Link ;
};
```

Solution:

```
struct Customer
{
    int CNo ;
    char CName[20] ;
    Customer *Link ;
};

class Queue
{
    Customer *front,*rear;
public:
    Queue( )
    { front=rear=NULL; }
    void Insert( );
    void Delete( );
```



```

void Display( );
};
void Queue::Delete( )
{ Customer *Temp;
if(front= =NULL)
cout<<"Queue Underflow. No element to delete";
else
{ cout<<"\n The customer number for the element to
delete.<<front->CNo;
cout<<"\n The customer name for the element to
delete"<<front->CName;
Temp=front;
front = front->Link;
delete Temp;
}
}

```

- 13) **Evaluate** the following **postfix** notation of expression : 15 3 2 + / 7 + 2. *

Solution:

step	Input symbols	Action	Stack status	Intermediate output
1	15	Push	15	
2	3	Push	15 3	
3	2	Push	15 3 2	
4	+	Pop twice(Binary operator)	15	3+2=5
5	5	Push	15 5	
6	/	Pop twice(Binary operator)		15/5=3
7	7	Push	3 7	
8	+	Pop twice(Binary operator)		3+7=10
9	10	Push	10	
10	2	Push	10 2	
11	*	Pop twice (Binary operator)		10*2=20
12	20	Push	20	
		No More Element		20 result