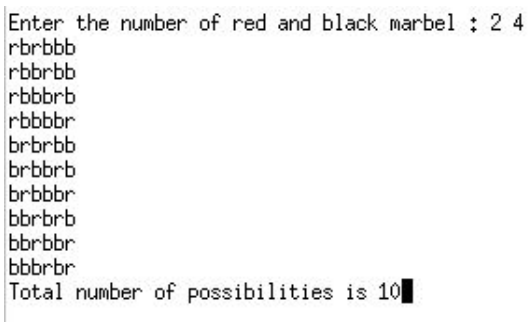
1 (a).

```
#include <iostream>
using namespace std;
int count=0;
void PrintOptions(int r,int b,int clr,string str){
	if(r<0 || b<0)
		return;
	if(r==0 && b==0){
		cout<<str<<endl;
		count++;
		return;
	}
	else{
		if(clr==0){
			if(r>0)
				PrintOptions(r-1,b,1,str+'r');
			if(b>0)
				PrintOptions(r,b-1,0,str+'b');
		}
		else{
			if(b>0)
				PrintOptions(r,b-1,0,str+'b');
		}
	}
}

int main() {
	int r,b;
	cout<<"Enter the number of red and black marbel : ";
	cin>>r>>b;
	PrintOptions(r,b,0,"");
	cout<<"Total number of possibilities”;
	cout<<count;
	return 0;
}
```

OUTPUT:



1(b):

```
#include <iostream>
using namespace std;
int count=0;
void PrintOptions(int r,int b,int clr,string str){
	if(r<0 || b<0)
		return;
	if(r==0 && b==0){
		cout<<str<<endl;
		count++;
		return;
	}
	else{
		if(clr==0){
			if(r>0) PrintOptions(r-1,b,1,str+'r');
			if(b>0) PrintOptions(r,b-1,0,str+'b');
		}
		else if(clr==1){
			if(r>0) PrintOptions(r-1,b,2,str+'r');
			if(b>0) PrintOptions(r,b-1,0,str+'b');
		}
		else if(clr==2){
			if(b>0) PrintOptions(r,b-1,0,str+'b');
}}}
int main() {
	int r,b;
	cout<<"Enter the number of red and black marbel : ";
	cin>>r>>b;
	PrintOptions(r,b,0,"");
	cout<<"Total number of
		possibilities is "<<count;
	return 0;
}
```

OUTPUT:

```
Enter the number of red and black marbel : 2 4
rrbbbb
rbrbbb
rbbrbb
rbbbrb
rbbbbr
brrbbb
brbrbb
brbbrb
brbbbr
bbrrbb
bbrbrb
bbrbbr
bbbrrb
bbbrbr
bbbbrr
Total number of possibilities is 15
```

2:

```
#include <bits/stdc++.h>
using namespace std;

vector<int> vt;
set<int> output;

int main() {
	int n,k;
	cout<<"Enter the size of the array and value of k : ";
	cin>>n>>k;
	vt.resize(n);
	for(int i=0;i<n;i++)
	{
		cin>>vt[i];
		if(i<k)
			output.insert(vt[i]);
		else
		{
			if(vt[i]>*output.begin())
			{
				output.insert(vt[i]);
				output.erase(output.begin());
			}
		}
	}
	cout<<k<<"th max value is: "<<*output.begin();
	return 0;
}
```

OUTPUT:

```
Enter the size of the array and value of k : 20 9
57 50 21 13 32 1 9 45 26 17 43 29 17 8 12 35 98 52 78 93
9th max value is: 35
```

3:

```
#include <bits/stdc++.h>
using namespace std;
vector<int> vt,output;

int main() {
	int n,k;
	cout<<"Enter the size of the array and value of k : ";
	cin>>n>>k;
	vt.resize(n);
	for(int i=0;i<n;i++)
	{
		cin>>vt[i];
		if(i<k)
			output.push_back(vt[i]);
		else if(i==k)
			sort(output.begin(),output.end());
		else
		{
			if(vt[i]>output[0] && vt[i]<output[k-1])
			{
				output[k-1]=vt[i];
				sort(output.begin(),output.end());
			}
		}
	}
	cout<<k<<"th smallest value is : "<<output[k-1];
	return 0;
}
```

OUTPUT:

```
Enter the size of the array and value of k : 20 7
67 54 22 16 32 1 9 45 26 17 43 29 19 8 12 35 98 52 78 83
7th smallest value is : 19
```

4:

```
#include <bits/stdc++.h>
using namespace std;

map<int,pair<int,int>> mp;
vector<int> GreedyJob;
int maxProf=0;
int main(){
	GreedyJob.resize(100);
	int jobId,Profit,Deadline;
	while(cin>>jobId>>Profit>>Deadline){
		mp.insert(make_pair(Profit,make_pair(Deadline,jobId)));
	}
	for(auto it=mp.rbegin();it!=mp.rend();it++)
	{
		pair<int,pair<int,int>> p=*it;
		Profit=p.first;
		pair<int,int> tmpPair=p.second;
		Deadline=tmpPair.first;
		jobId=tmpPair.second;
		for(int i=Deadline;i>=1;i--)
		{
			if(GreedyJob[i]==0)
			{
				GreedyJob[i]=jobId;
				maxProf+=Profit;
				break;
			}
		}
	}
	for(int i=1;i<GreedyJob.size();i++)
		if(GreedyJob[i])
			cout<<char(GreedyJob[i]+'A'-1)<<" ";
		cout<<endl<<"Max Profit is: "<<maxProf;
		return 0;
	}
```

```
1 10 2
2 19 1
3 77 2
4 35 2
5 15 3
```

Time(sec) : 0

Output:

```
D C E
Max Profit is: 127
```