1. What will be the final outcome of the following complexity?

O(N\*N+N+N/2)

* 1. O(N)
  2. O(N/2)
  3. **O(N\*N)**
  4. O(N\*N+N)

**Explanation:**  আমরা জানি টাইম কমপ্লেক্সিটি এর ক্ষেত্রে আমরা সবসময় সবচেয়ে বড় ফ্যাক্টর নিয়ে চিন্তা করবো। এক্ষেত্রে N\*N , N , N/2 => N এর মধ্যে সবচেয়ে worst কমপ্লেক্সিটি হলো N\*N . তাই এর টাইম কমপ্লেক্সিটি O(N\*N)

1. What will be the time complexity of the following loop?

for(int i=0;i<N/2;i++)

* 1. **O(N)**
  2. O(N\*N)
  3. O(Nlog(N))
  4. O(log(N))

**Explanation:**  আপাতদৃষ্টিতে এই কোডের টাইম কমপ্লেক্সিটি O(N/2 ) মনে হতে পারে কারণ ০ থেকে শুরু করে N/2 পর্যন্ত লুপ চলছে । কিন্তু আমরা জানি টাইম কমপ্লেক্সিটি এর ক্ষেত্রে আমরা constant factor টুকু বাদ দিতে পারি। সেক্ষেত্রে এই কোডের টাইম কমপ্লেক্সিটি হবে O(N).

1. What will be the time complexity of the following loop?

for(int i=1;i<=N;i++) {

for(int j=1;j<=M;j++){

}

}

* 1. O(N)
  2. **O(N\*M)**
  3. O(N\*i)
  4. O(log(N))

**Explanation:** উক্ত কোডের ক্ষেত্রে , কোডটি একটি নেস্টেড লুপের। এর ভিতরের লুপটি ১ থেকে শুরু করে M পর্যন্ত অর্থাৎ M বার চলছে যার টাইম কমপ্লেক্সিটি M . এর বাইরের লুপটি ১ থেকে শুরু করে N পর্যন্ত চলছে। তাই এই কোডের টাইম কমপ্লেক্সিটি N . যেহেতু এটি একটি নেস্টেড লুপ , তাই আমরা জানি নেস্টেড লুপের ক্ষেত্রে বাইরের লুপের টাইম কমপ্লেক্সিটি এর সাথে ভিতরের লুপের টাইম কমপ্লেক্সিটি গুণ হয় । সুতারাং উক্ত কোডের টাইম কমপ্লেক্সিটি O(N\*M)

1. What will be the time complexity of the following loop?

for(int i=1;i<=N;i\*=10)

* 1. O(N)
  2. O(N\*N)
  3. O(N/10)
  4. **O(log(N))**

**Explanation:** এই কোডের ক্ষেত্রে ইনক্রিমেন্ট এর পার্ট টুকু তে i এর মান প্রতিবার ১০ এর সাথে গুণ হচ্ছে । আমরা জানি , যদি আমরা দেখি কোনো একটি কোডের পার্ট কোনো একটি মান দ্বারা গুণ / ভাগ হচ্ছে সেই কোডের টাইম কমপ্লেক্সিটি লগারিদমিক হয়ে থাকে।

সুতারাং উক্ত কোডের টাইম কমপ্লেক্সিটি O(log(N))

1. What will be the time complexity of the following loop?

for(int i=1;i\*2<=N;i++)

* 1. **O(N)**
  2. O(N/2)
  3. O(sqrt(N))
  4. O(log(N))

**Explanation:** কন্ডিশনের পার্টটুকু যদি আমরা সাজিয়ে লিখি তাহলে দেখা যায় , i\*2<=N **->** i<=N/2 . অর্থাৎ লুপটি ১ থেকে শুরু করে N/2 বার চলছে। সুতারাং এই কোডের টাইম কমপ্লেক্সিটি O(N)

1. What will be the time complexity of the following loop?

for(int i=1;i\*i<=N;i++)

* 1. O(N)
  2. O(N\*N)
  3. **O(sqrt(N))**
  4. O(log(N))

**Explanation:** sqrt টাইম কমপ্লেক্সিটি এর মডিউলে এই কোডের টাইম কমপ্লেক্সিটি সম্পর্কে দেখানো হয়েছে । উক্ত কোডের টাইম কমপ্লেক্সিটি O(sqrt(N)).

1. What will be the time complexity of the following loop?

#include<bits/stdc++.h>

using namespace std;

int main()

{

int n; cin>>n;

int a[n];

for(int i=0;i<n;i++)

{

cin>>a[i];

}

sort(a,a+n);

for(int i=0;i<n;i++)

{

cout<<a[i]<<" ";

}

return 0;

}

1. O(N)
2. O(N\*N)
3. **O(Nlog(N))**
4. O(log(N))

**Explanation:** উক্ত কোডের ক্ষেত্রে , প্রথম লুপের টাইম কমপ্লেক্সিটি O(N) , এরপর বিল্ট ইন sort function ব্যবহার করা হয়েছে যার টাইম কমপ্লেক্সিটি O(NlogN) , এরপরে প্রিন্টিং এর লুপটির টাইম কমপ্লেক্সিটি O(N). টোটাল টাইম কমপ্লেক্সিটি : O(N + NlogN + N)। যার মধ্যে bigger time complexity হলো O(NlogN) সুতারাং উক্ত কোডের টাইম কমপ্লেক্সিটি O(NlogN).

1. What will be the space complexity of the following loop?

#include<bits/stdc++.h>

using namespace std;

int main()

{

int n; cin>>n;

int a[n];

for(int i=0;i<n;i++)

{

cin>>a[i];

}

sort(a,a+n);

for(int i=0;i<n;i++)

{

cout<<a[i]<<" ";

}

return 0;

}

1. **O(N)**
2. O(N\*N)
3. O(Nlog(N))
4. O(log(N))

**Explanation:** উক্ত কোডের ক্ষেত্রে , N এর মানের উপর dependent অর্থাৎ N সাইজের একটি array ডিক্লেয়ার করা হয়েছে সুতারাং উক্ত কোডের স্পেস কমপ্লেক্সিটি O(N)

1. What will be the time complexity of the following loop?

#include<bits/stdc++.h>

using namespace std;

int main()

{

int n; cin>>n;

int a[n];

for(int i=0;i<n;i++) {

cin>>a[i];

}

for(int i=0;i<n-1;i++) {

for(int j=i+1;j<n;j++) {

if(a[i]>a[j]) swap(a[i],a[j]);

}

}

for(int i=0;i<n;i++) {

cout<<a[i]<<" ";

}

return 0;

}

1. O(N)
2. **O(N\*N)**
3. O(Nlog(N))
4. O(log(N))

**Explanation:** উক্ত কোডের ক্ষেত্রে , কোডটি একটি নেস্টেড লুপের। এর ভিতরের লুপটি i+১ থেকে শুরু করে N পর্যন্ত অর্থাৎ N বার চলছে যার টাইম কমপ্লেক্সিটি N . এর বাইরের লুপটি 0 থেকে শুরু করে N-1 পর্যন্ত চলছে। তাই এই কোডের টাইম কমপ্লেক্সিটি N . যেহেতু এটি একটি নেস্টেড লুপ , তাই আমরা জানি নেস্টেড লুপের ক্ষেত্রে বাইরের লুপের টাইম কমপ্লেক্সিটি এর সাথে ভিতরের লুপের টাইম কমপ্লেক্সিটি গুণ হয় । সুতারাং উক্ত কোডের টাইম কমপ্লেক্সিটি O(N\*N)

1. What will be the time complexity of the following loop?

#include<bits/stdc++.h>

using namespace std;

int main()

{

int n; cin>>n;

for(int i=0;i<n-1;i++) {

for(int j=n;j>=1;j/=2) {

cout << i << “ “ << j << endl ;

}

}

return 0;

}

1. **O(N\*logN)**
2. O(N\*N)
3. O(sqrtN)
4. O(log(N))

**Explanation:** উক্ত কোডের ক্ষেত্রে , কোডটি একটি নেস্টেড লুপের। এর ভিতরের লুপটি N থেকে শুরু করে 1 পর্যন্ত চলছে এবং প্রতিক্ষেত্রে i এর মান ২ দ্বারা ভাগ হয়ে ছোট হয়ে যাচ্ছে। আমরা জানি , increment এর পার্ট কোনো একটি মান দ্বারা গুণ হয়ে বাড়লে / ভাগ হয়ে ছোট হলে তার টাইম কমপ্লেক্সিটি O(logN). এর বাইরের লুপটি 0 থেকে শুরু করে N-1 পর্যন্ত চলছে। তাই এই কোডের টাইম কমপ্লেক্সিটি N . যেহেতু এটি একটি নেস্টেড লুপ , তাই আমরা জানি নেস্টেড লুপের ক্ষেত্রে বাইরের লুপের টাইম কমপ্লেক্সিটি এর সাথে ভিতরের লুপের টাইম কমপ্লেক্সিটি গুণ হয় । সুতারাং উক্ত কোডের টাইম কমপ্লেক্সিটি O(N\*logN)