### 一、编程题（2道\*25分）

1、输入一个整数 n，输出从 1 到 n 的所有素数。

2、输入多个整数，直到输入 -1，计算所有偶数的总和。

Int n

While(1){

Scanf(“%d”,&n)

If(n==-1) break;

Else if (n%2!=0)

}

### 二、程序分析题（2道\*20分）

3、分析以下 C 代码的错误及可能的输出：

#include <stdio.h>

int main() {

int arr[3] = {1, 2, 3};

printf("%d", arr[3]);

//会发生越界，数组长度为3，仅可以访问a[0],a[1],a[2]

//可能返回一个内存垃圾值，或是段错误，取决于编译器机制和访问内存

return 0;

}

4、解释 TCP 三次握手的流程，并分析如果第三次握手失败的影响。

### 三、简答题（1道\*10分）

5、SQL 注入的攻击原理及防范措施。

任何于数据库交互的地方均可能出现sql注入，其原理就是利用非法的sql语句获取数据库的敏感信息。

1、

#include <stdio.h>

int is\_prime(int n) {

if (n < 2) return 0;

for (int i = 2; i \* i <= n; i++) {

if (n % i == 0) return 0;

}

return 1;

}

int main() {

int n;

scanf("%d", &n);

for (int i = 2; i <= n; i++) {

if (is\_prime(i)) printf("%d ", i);

}

printf("\n");

return 0;

}

2、

#include <stdio.h>

int main() {

int num, sum = 0;

while (1) {

scanf("%d", &num);

if (num == -1) break;

if (num % 2 == 0) sum += num;

}

printf("%d\n", sum);

return 0;

}

3、

该代码访问了数组越界位置，可能会打印未知值或导致运行时错误。

4、

若第三次握手失败，服务器不会建立连接，可能会重试或超时

5、

SQL 注入通过构造恶意 SQL 语句攻击数据库，应使用参数化查询防御。