<table>
<thead>
<tr>
<th>No.</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Start of the text</td>
</tr>
<tr>
<td>2</td>
<td>Paragraph 1</td>
</tr>
<tr>
<td>3</td>
<td>Paragraph 2</td>
</tr>
<tr>
<td>4</td>
<td>Paragraph 3</td>
</tr>
<tr>
<td>5</td>
<td>Paragraph 4</td>
</tr>
<tr>
<td>6</td>
<td>Continued paragraph 4</td>
</tr>
</tbody>
</table>

**Guli Nam 3:**

[Text content not transcribed]

**Guli Nam 4:**

[Text content not transcribed]
```c
#include <stdio.h>

int main() {
    int x = 5;
    int y = 3;
    int z = x * y;
    printf("The result is %d\n", z);
    return 0;
}
```
Click here for the full text.
```c
void bag_copy(bag *dest, bag *src)
{
    dest->size = src->size;
    dest->capacity = src->capacity;
    dest->elements = (int *) malloc(dest->capacity * sizeof(int));
    if (dest->elements == NULL)
        return;
    for (int i = 0; i < dest->size; i++)
        dest->elements[i] = src->elements[i];
}

int bag_remove(bag *bag)
{
    int num_removed = 0;
    for (int i = 0; i < bag->size;)
    {
        if (bag->elements[i] < 0)
        {
            num_removed++;
            i++;
        }
        else
            i++;
    }
    if (num_removed > 0)
        bag->size -= num_removed;
    return num_removed;
}

int bag_max(bag *bag)
{
    int max = INT_MIN;
    for (int i = 0; i < bag->size;)
    {
        if (bag->elements[i] > max)
            max = bag->elements[i];
        i++;
    }
    return max;
}

int bag_min(bag *bag)
{
    int min = INT_MAX;
    for (int i = 0; i < bag->size;)
    {
        if (bag->elements[i] < min)
            min = bag->elements[i];
        i++;
    }
    return min;
}
```

```java
{
    printf("%s", line);
    if (strcmp(line, name) == 0)
        printf("match found!
"");
}

int main()
{
    printf("Hello, World!
"");
    return 0;
}
```
```cpp
int main()
{
    // Your code here
    return 0;
}
```
C++ 2. TIME

```c++
#include <iostream>

int main() {
    int a = 10,
        b = 5;

    std::cout << "a = \"" << a << ", b = \"" << b << std::endl;

    return 0;
}
```

C-SHELL 2. TIME

```bash
#!/bin/bash

a=10
b=5

echo "a = $a, b = $b"
```

C++ Program:

1. This program demonstrates the use of variables in C++.
2. It declares two variables `a` and `b` and assigns them the values 10 and 5, respectively.
3. The `std::cout` statement is used to print the values of `a` and `b` to the console.

C-SHELL Program:

1. This script is written in C-SHELL, a shell scripting language.
2. It assigns values to variables `$a` and `$b` and prints them.

```cpp
int main()
{
    // Code goes here...
}
```
The output of the program is:

```c

```
C. Shell: 2 Tkw

```c
int main(void)
{
    // Code goes here
    return 0;
}
```

**Notes:**
- The code snippet is a simple C program that prints `main` and returns 0.
- It appears to be a part of a larger piece of code, possibly for testing or demonstration purposes.
- The comments are in English, indicating that the code is written in English.
- The code structure is typical of a C program, with function definitions and variable declarations.

**Potential Issues:**
- The file name or directory path is not specified, which might lead to confusion when attempting to compile or run the code.
- The context of the code is not clear from the snippet alone, suggesting that more information might be needed to fully understand its purpose.
- The code snippet is short and does not provide enough context to assess its functionality or correctness.

If you have more context or additional code, please provide it for a more detailed analysis.
```cpp
#include <iostream>

// Function to check if a number is prime
bool isPrime(int n) {
    if (n <= 1) return false;
    for (int i = 2; i <= n / 2; ++i) {
        if (n % i == 0) return false;
    }
    return true;
}

int main() {
    int num;
    std::cout << "Enter a number: " << std::endl;
    std::cin >> num;

    if (isPrime(num)) {
        std::cout << num << " is a prime number.\n";
    } else {
        std::cout << num << " is not a prime number.\n";
    }

    return 0;
}
```

There is an error in the test form.
```java

// Table 2B

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```
C++: 3 TIPs

1. In the book, find the Ten 3 TIPs. This problem:
   # the text is not correct if it is a note of a book that does not appear
   # in the book:

```cpp
#include <iostream>
#include <vector>

int main()
```

2. Books are fun. But, the more complicated:
   # we could have avoided the books and read only.
   # with the book, only one of your book choices:

```cpp
int main()
```

3. Once the book names are not correct, the book names are:
   # more complicated.

```cpp
int main()
```
null
```c
5. (++) :<  

else if (number > 0) 
    { 
        // do something 
    } 
else if (number < 0) 
    { 
        // do something else 
    } 
else 
    { 
        // do nothing 
    }

// end of function
```