Name:	Roll No:	

Programming Language Concepts

Quiz 2, II Semester, 2024–2025

13 February, 2025

1. Consider the following Rust functions.

(i)
```
fn quiz1a(){
    let s1 = "hello";
    let s2 = "world";
    let a = [s1,s2];
    let head = a[0];
    let b = (head == a[0]);
    println!("{}",b)
}
```

(ii)
```
fn quiz1b(){
    let s1 = String::from("hello");
    let s2 = String::from("world");
    let a = [s1,s2];
    let head = a[0];
    let b = (head == a[0]);
    println!("{}",b)
}
```

(iii)
```
fn quiz1c(){
    let s1 = String::from("hello");
    let s2 = String::from("world");
    let a = [s1,s2];
    let head = &a[0];
    let b = (head == a[0]);
    println!("{}",b)
}
```

(iv)
```
fn quiz1d(){
    let s1 = String::from("hello");
    let s2 = String::from("world");
    let a = [s1,s2];
    let head = &a[0];
    let b = (head == &a[0]);
    println!("{}",b)
}
```

Fill in each entry in the following table with **Yes** or **No**.

	Compiles	*Runs*
`quiz1a`		
`quiz1b`		
`quiz1c`		
`quiz1d`		

. . . Question 2 on reverse

2. Consider the following Rust functions.

(i)
```
fn quiz2a(){
    let mut s = String::from("PLC 2025");
    let a = &s[..4];
    let b = &s[4..];
    s = String::from("Hello world");
}
```

(ii)
```
fn quiz2b(){
    let mut s = String::from("PLC 2025");
    let a = &s[..4];
    let b = &s[4..];
    s = String::from("Hello world");
    println!("s:{}, a:{}, b:{}",s,a,b);
}
```

(iii)
```
fn quiz2c(){
    let mut s = String::from("PLC 2025");
    let a = &mut s[..4];
    let b = &mut s[4..];
    s = String::from("Hello");
}
```

(iv)
```
fn quiz2d(){
    let mut s = String::from("PLC 2025");
    let a = &mut s[..4];
    let b = &mut s[4..];
    s = String::from("Hello");
    println!("s:{}, a:{}, b:{}",s,a,b)
}
```

Fill in each entry in the following table with **Yes** or **No**.

	Compiles	*Runs*
`quiz2a`		
`quiz2b`		
`quiz2c`		
`quiz2d`		