Yohandi - 120040025

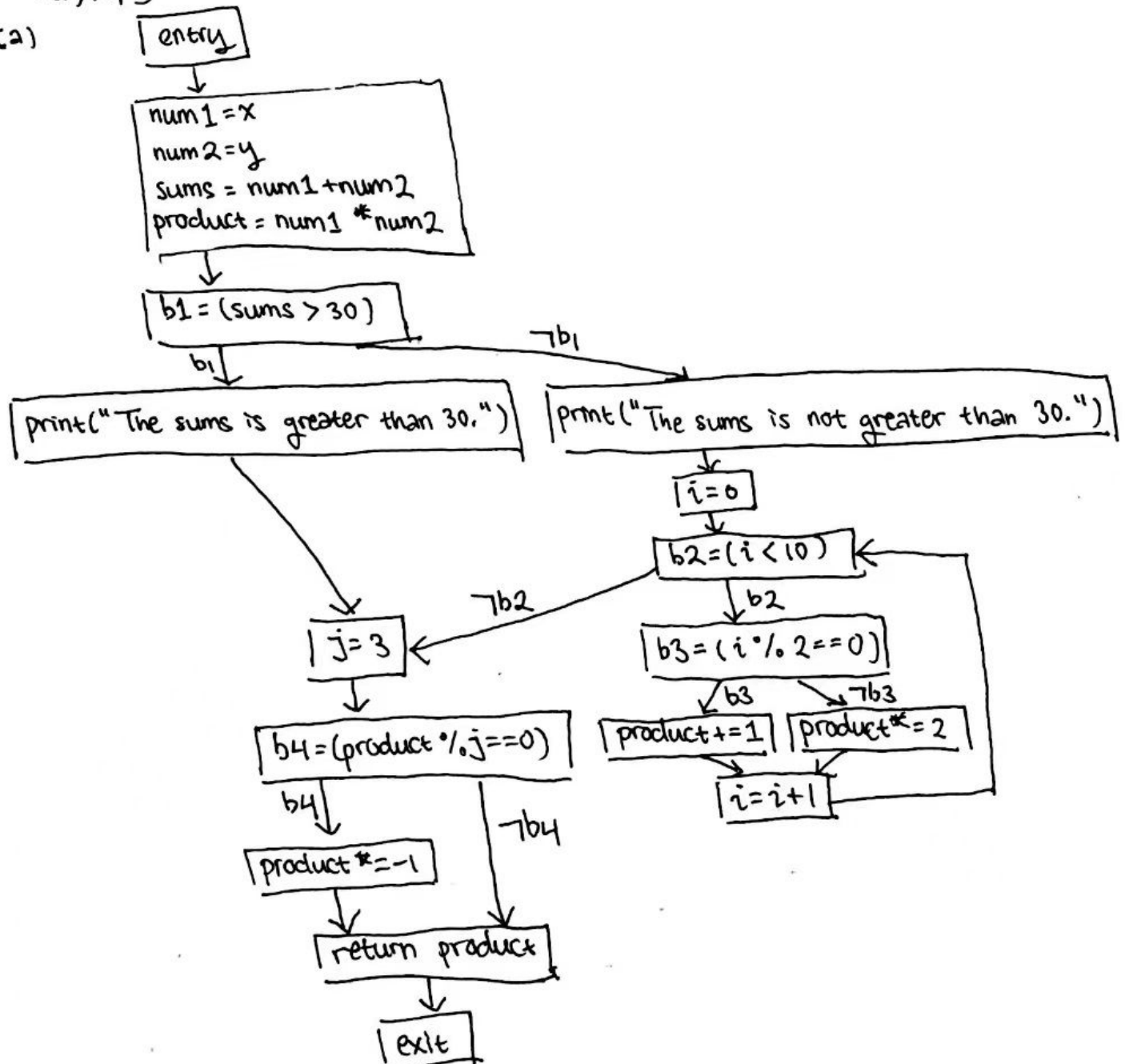CSC4001 Assignment 1

1.1 (a) C

(b) A, B, D

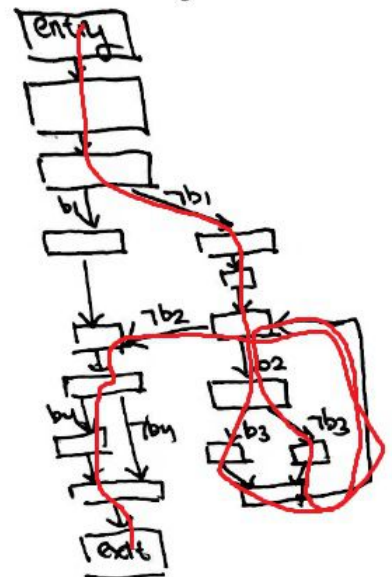(c) C, D

1.2 (a) D

(b) C, D

(c) C

(d) A, B

2 (a)

```
entry
   ↓
num1 = x
num2 = y
sums = num1 + num2
product = num1 * num2
   ↓
b1 = (sums > 30)
```

```
b1 → print("The sums is greater than 30.")
¬b1 → print("The sums is not greater than 30.")
```

```
i = 0
   ↓
b2 = (i < 10)
   ↓ b2
b3 = (i % 2 == 0)
   b3 ↓        ↓ ¬b3
product += 1   product *= 2
        ↓      ↓
       i = i + 1
```

```
¬b2 → j = 3
   ↓
b4 = (product % j == 0)
   b4 ↓        ↓ ¬b4
product *= -1
        ↓      ↓
     return product
        ↓
      exit
```

(b) Yes, at least two tests are needed, example:

→ x = 30, y = 1

→ x = 1, y = 1
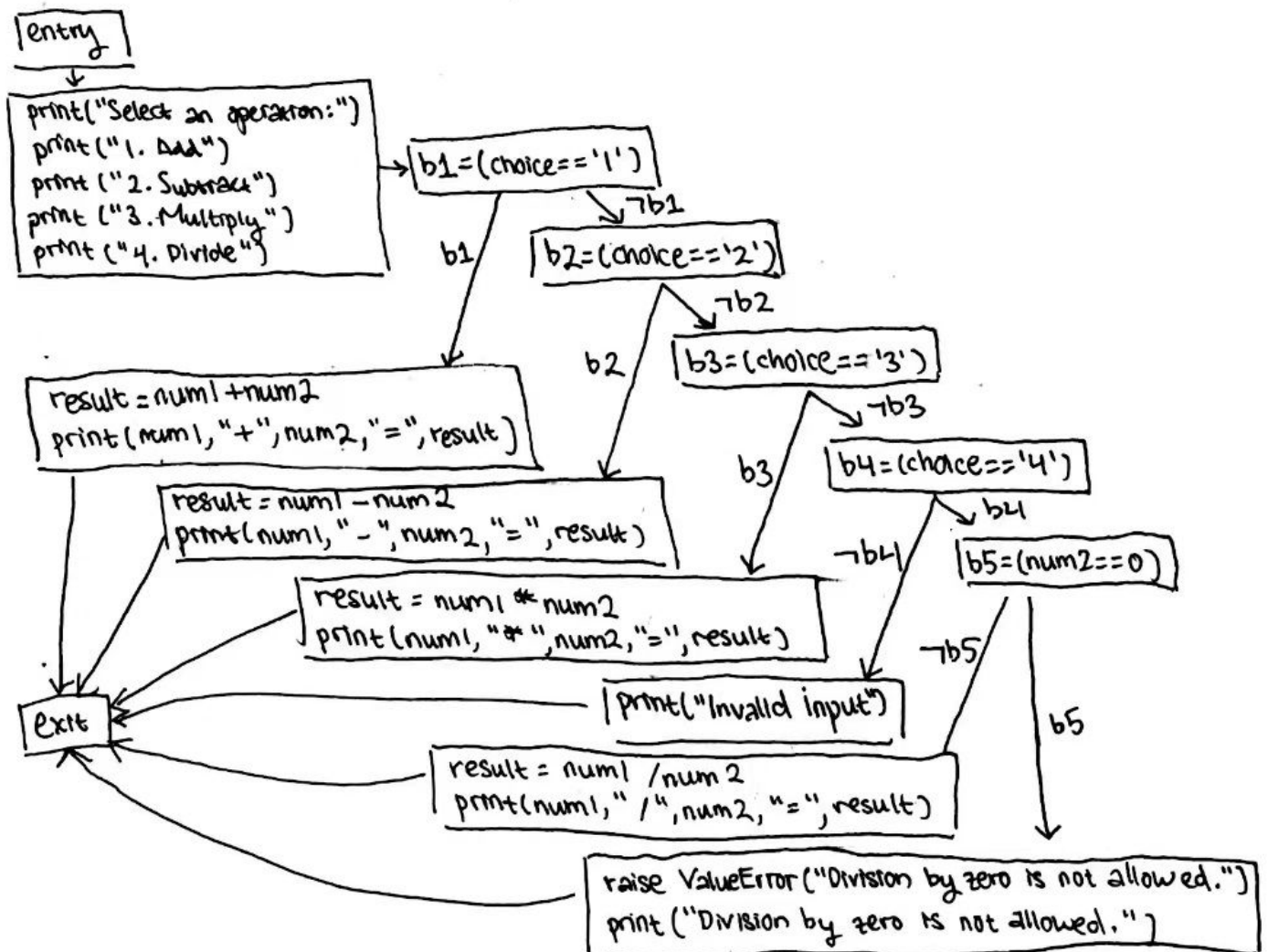
(c) Yes, at least four tests are needed, example:

→ x = 30, y = 1   → x = 1, y = 1   (note that: this is with an assumption that the for
→ x = 31, y = 1   → x = 2, y = 1   loop (acyclic) part is considered as a
                                        long and fixed path that it always cover
                                        the "only" path which is i = 0, ..., 9)

X=30, y=1

entry, b₁, b₄, exit

X=31, y=1

entry, b₁, ¬b₄, exit

X=1, y=1

entry, ¬b₁, b₂, b₃, b₂, ¬b₃, b₂, b₃, b₂, ¬b₃, b₂, b₃, b₂, ¬b₃, b₂, b₃, b₂, ¬b₃, b₂, b₃, b₂, ¬b₃, ¬b₄, exit

X=2, y=1

entry, ¬b₁, b₂, b₃, b₂, ¬b₃, b₂, b₂, b₂, ¬b₃, b₂, b₃, b₂, ¬b₃, b₂, b₃, b₂, ¬b₃, b₄, exit

3

```
entry
  ↓
print("Select an operation:")
print("1. Add")
print("2. Subtract")        →  b1=(choice=='1')
print("3. Multiply")                       ↘ ¬b1
print("4. Divide")              b2=(choice=='2')
                          b1                    ↘ ¬b2
                                          b3=(choice=='3')
                               b2                  ↘ ¬b3
result = num1+num2                        b3   b4=(choice=='4')
print(num1,"+",num2,"=",result)                      ↘ b4
                                                 b5=(num2==0)
       result = num1−num2
       print(num1," − ",num2,"=",result)
                                    ¬b4
          result = num1 * num2
          print(num1,"*",num2,"=",result)          ¬b5
                                                          b5
exit ←        print("Invalid input")

            result = num1 /num2
            print(num1," /",num2,"=",result)

              raise ValueError("Division by zero is not allowed.")
              print("Division by zero is not allowed.")
```

(a) Statement coverage: $\frac{18}{21} \approx 86\%$

   branch coverage: $\frac{8}{10} = 80\%$

   path coverage: $\frac{4}{6} \approx 67\%$

(b) Test set that achieve 100% statement coverage:
   {choice ='1', num1 =1, num2 =1};
   {choice='2', num1=2, num2=2};
   {choice ='3', num1 =3, num2 =3};
   {choice ='4', num1 =4, num2 =4};
   {choice ='4', num1 =4, num2 = 0};
   {choice ='5', num1 =5, num2 =5}.

   Test set that achieve 100% branch coverage:
   {choice='1', num1 =1, num2=1};
   {choice ='2', num1 = 2, num2= 2};
   {choice ='3', num1 =3, num2=3};
   {choice ='4', num1 = 4, num2 =4};
   {choice ='4', num1=4, num2=0};
   {choice ='5', num1 =6, num2= 9}.

   Test set that achieve 100% path coverage:
   {choice='1', num1=1, num2=1};
   {choice ='2', num1 =2, num2 = 2};
   {choice ='3', num1= 3, num2 =3};
   {choice ='4', num1 = 4, num2= 4};
   {choice = '4', num1 = 4, num2= 0};
   {choice ='5', num1 =42, num2=0}.