**Progress test 1**

**CSI104**

|  |  |
| --- | --- |
| **Student Name:** | **Huỳnh Ngọc Quang** |
| **Student ID:** | **SE181838** |

**Q1.** Convert the following hexadecimal number to decimal without using a calculator, showing your work in step by step: **(2 marks)**

**BCxx**

*xx: xx is the two last number of your student id*

Answer: (BC38)16 = (B x 163 + C x 162 + 3 x 161 + 8 x 160)10

= (11 x 163 + 12 x 162 + 3 x 161 + 8 x 160)10

= (45056 + 3072 + 48 + 8 )10

= (48184)10

**Q.2** Convert the following decimal numbers to binary and hexadecimal without using a calculator, showing your work in step by step: **(2 marks)**

**xxx**

*xxx: xxx is the three last number of your student id*

*Answer:*

(838)10 = (512 + 326)10

= (512 + 256 + 70)10

= (512 + 256 + 64 + 4 + 2)10

= (29 + 28 + 26 + 22 + 21)10

= (00000011 01000110)2

Mỗi số hexa bằng 4 bit binary, nên chúng ta có thể tiến hành gom 4 bit của binary thành 1 số hexa.

(838)10= (0000 0011 0100 0110)2

= ( 0 3 4 6 )16

= (346)16

**Q3.** Change the following decimal numbers to 8-bit two’s complement integers (**2 marks**)

**11x** (where x is the last number of your student id)

**2xx** (where x is the two last number of your student id)

*Answer:*

(118)10

Chuyển đổi theo thông thường:

(118)10 = (127 - 9)10

= (127 – 8 - 1)10

= (01110110)2

Phương pháp bù 2 dùng cho số âm, nhưng ở đây là số dương nên chúng ta không cần phải đảo bit và cộng thêm 1.

(238)10 không thể chuyển sang hệ nhị phân theo phương pháp bù 2 dùng 8 bit.

Vì chỉ có 8 bit, nếu theo phương pháp thông thường lưu trữ không dấu thì ta có thể lưu trữ giá trị từ 0 – 28 (0 - 256) tuy nhiên để biểu diễn số âm, ta dùng 1 bit đầu để quy định dấu nên ta chỉ có thể lưu trữ giá trị trong khoảng từ -27 đến 27 – 1 (-128 đến 127).

Và vì 238 > 127 nên đã vượt ra ngoài phạm vi lưu trữ nên chúng ta không thể biểu diễn 238 sang hệ nhị phân theo phương pháp bù 2 dùng 8 bit.

**Q4.** Show the Excess\_127 (single precision) representation of the decimal number: **(2 marks)**

-**1xx.xx**

Where xx is the two last number of your student id

*Answer:*

(-118.38)10

Chuyển sang hệ nhị phân ta được

(118)10 = (127 - 9)10

= (127 – 8 - 1)10

= (01110110)2

0.38 x 2 = 0.76

0.76 x 2 = 1.52

0.52 x 2 = 1.04

0.04 x 2 = 0.08

0.08 x 2 = 0.16

0.16 x 2 = 0.32

0.32 x 2 = 0.64

0.64 x 2 = 1.28

0.28 x 2 = 0.56

0.56 x 2 = 1.12

0.12 x 2 = 0.24

0.24 x 2 = 0.48

0.48 x 2 = 0.96

0.96 x 2 = 1.92

0.92 x 2 = 1.84

0.84 x 2 = 1.68

0.68 x 2 = 1.36

0.36 x 2 = 0.72

0.72 x 2 = 1.44

0.44 x 2 = 0.88

0.88 x 2 = 1.76

(118.38)10 = (01110110.011000010100011110101)2

= (1.110110011000010100011110101 x 26)2

Sign: 1

Exponent: (6 + 127)10 = (128 + 5)10 = (10000101)2

Mantisa: 110110011000010100011110101

Vì mantisa chỉ có 23 bit nên chúng ta loại bỏ phần thừa

Mantisa: 11011001100001010001111

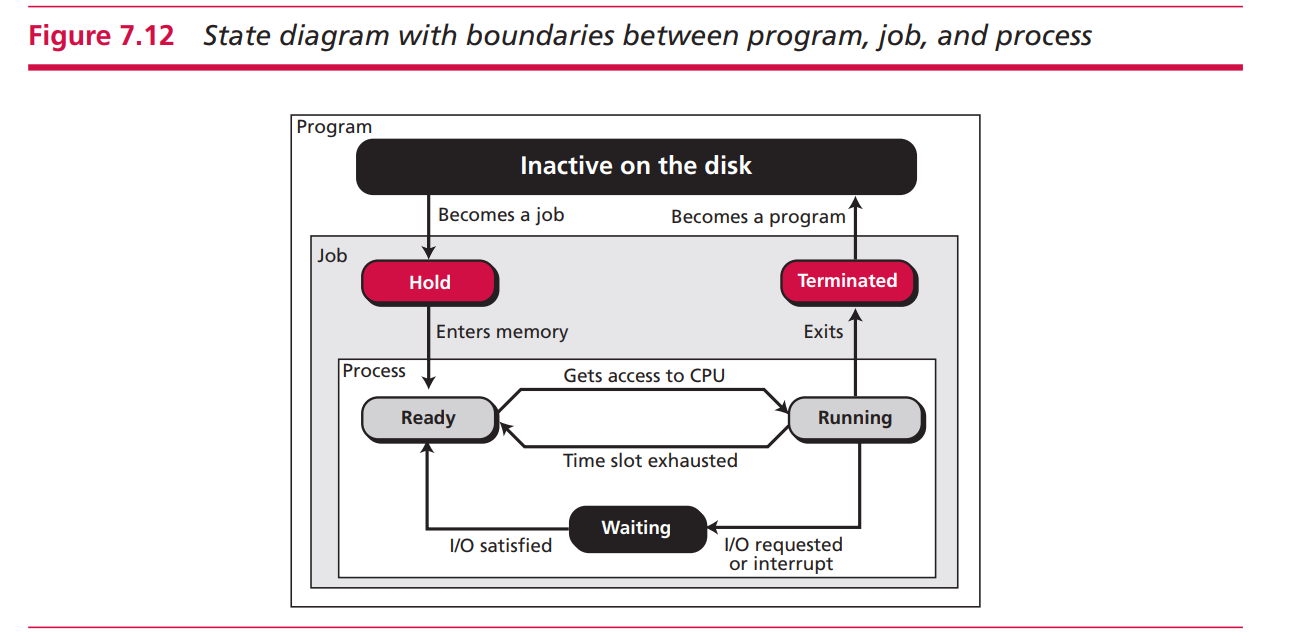
Biểu diễn của (-118.38)10 ta được

11000010111011001100001010001111

**Q.5** Explain in your words: (**2 marks**)

**Figure 7.12, Algorithm 8.5**

*Answer:*



Mọi chương trình ban đầu đều sẽ được lưu và không hoạt động trên ổ cứng ở mức này thì nó được gọi là 1 chương trình (program). Sau khi được người dùng ở chương trình này lên thì chương trình sẽ được vào hang đợi (Job) chuẩn bị khởi chạy. Khi chương trình bắt đầu được khởi chạy, dữ liệu sẽ được nạp vào trong bộ nhớ (memory) và bây giờ chương trình sẽ là 1 tiến trình (process). Khi chương trình đã sẵn sàng chạy thì nó sẽ giao tiếp với CPU, I/O module, Memory và theo thời gian, điều khiển (ngắt, mở) của CPU. Khi chương trình đang chạy mà gặp tín hiệu ngắt từ người dùng thì process sẽ trở lại thành Job vào hang đợi để giải phóng bộ nhớ. Sau khi hoàn tất đóng chương trình và giải phóng bộ nhớ thì chương trình sẽ quay về bản chất ban đầu chỉ là một chương trình nằm trên ổ cứng.

A screenshot of a math program

Description automatically generated with low confidence

Chương trình (hàm) trên sẽ nhận vào 1 tham số là 1 mảng list có nhiều hơn 1000 phần từ và sẽ trả về giá trị số nguyên nhỏ nhất có trong mảng trong khoảng 1000 phần đầu tiên.

Thuật toán:

Ban đầu mặc định số nhỏ nhất (smallest) là dương vô cùng, sau đó duyệt từ phần từ đầu đến phần tử thứ 1000 nếu có phần từ nào trong mảng nhỏ hơn phần tử nhỏ nhất (smallest) thì gán lại cho smallest bằng phần tử đó. Chương trình sẽ dùng biến count làm index cho mảng. Sau khi duyệt trong khoảng 1000 phần tử đầu tiên, smallest sẽ lưu trữ giá trị nhỏ nhất có trong mảng và hàm sẽ trả về giá trị nhỏ nhất này