**System program – Assignment1**

**SFP**

**시스템경영공학과 2016314726**

**정영준**

**1) int2sfp**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Int2sfp function gets integer type variable and return single float point variable**

**First check special cases: input is 0 or +inf or -inf**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Then, convert decimal to binary by continuous %.**

**If input became 0 break the loop.**

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

**Get rank of binary number and decimal number by while loop statement.**

**And newly Setting index using “count” variable that contains rank information.**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Last, while binary number is not 0, add (binary number \* rank) to frac.**

**And add frac parts \* index to result.**

**Return result**

**2) sfp2int**

텍스트이(가) 표시된 사진

자동 생성된 설명

**First, Check special cases: input is 0, +inf, -inf**

텍스트, 스크린샷, 화면이(가) 표시된 사진

자동 생성된 설명

**Find first appeared “1” and get its position.**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Move current position to 15 and going through all position to calculate frac parts.**

**Then, calculate exp parts.**

**exp = E + bias So, E = exp – bias**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Compare E and binary number for round-to-zero process.**

**Add (Binary number of each position \* its rank) to result.**

**Last, if remainder exists convert result to negative value.**

**And return result**

**3) float2sfp**

텍스트이(가) 표시된 사진

자동 생성된 설명

**First, check special cases that input is 0 or +inf or – inf.**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Initializing variable and move index to appropriate position.**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Initialize array for binary number to 0.**

**Check sign bits then convert “int\_num” to binary number.**

**Then save binary number’s data to b\_array.**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Decimal places parts.**

**Check if calculated input is Special Cases. Then, Process round-to-zero.**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Last, calculate exp (E + bias) and set rank of binary number using for statement.**

**And calculate frac parts, add it to result.**

**4) sfp2float**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Check input is negative or positive.**

**And Initialize all variable and array.**

**First, find frac parts position using value of (input % 2)**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Then, calculate exp parts. E = exp -bias**

**Decimal parts(소수점 아래) calculation parts. (for statement)**

**Last, if sign bit is 1 return -1 \* result.**

텍스트이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명

**Util function for sfp\_add function.**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Calculate Exp by util function(calExp).**

**And check special cases.**

**Then compare exp\_2 and exp\_1 for swapping two variable.**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Initialize variables parts.**

**And input1 and input2 calculation ( % 1024)**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Implemented using if statements to process differently depending on the sign bit of two number.**

**By various condition, frac is calculated differently.**

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트, 화면이(가) 표시된 사진

자동 생성된 설명

**Using util functions(checkRoundtoZero and checkLeftShiftPreCondition) to do round-to-zero process.**

**Last, add calculated frac to result. And check round variable to decide whether plus 1 to result or not.**

**6) sfp2bits**

텍스트이(가) 표시된 사진

자동 생성된 설명

**Initialize result array to 0.**

**If result % 2 is 1, assign ‘1’ to appropriate position in result array.**

**If result % 2 is 0, assign ‘0’ to appropriate position in result array.**

**After while loop return final\_result.**