Section: t/me(11-12)

Q1) (4 marks) A) Discuss two features that make Java better with respect to reliability than C++.

1. Java it have indexing and it will check if the index out of bound or not.
in c++ it have Pounter. it allows user to go out of bound.

2. Java it will transalit to interm adat langug
c++ it will transalit to machean langug.

2

B) (2 marks) Explain how the implementation of Java makes it better with respect to portability than C++.

Java it will transalit to interm adat langug
Be for transalit to machean langug
that will respect to Protability.
~~it will work w be effecnt work with~~
~~allot of Os.~~

2

C) (4 marks) Discuss two advantages of C++ compared to Java.

1. ~~easay to learn~~
   respect the cost

2. ~~easay to~~ ~~th some easy to~~
   ~~will be easy to learn~~
   easisir to learn.

Q2) (4 marks) a) Assuming that a float number consists of an optional integer part and an optional fraction part, write a regular expression that describes the language of all float numbers. Note that the strings "14.15", ".15" and "15." are all legal but the string "." is not.

dig = [0-9];

$(dig^{+} + \cdot)(dig)(dig^{*} + \cdot)$

b) (4 marks) Write a CFG that describes the language of float numbers as described above.

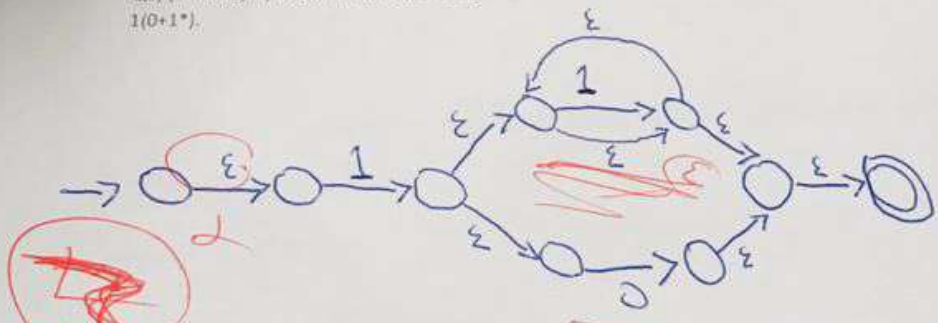
S → dig . dig | . dig | dig .

C) (2 marks) Consider a language defined over S={0,1} where a string belongs to the language if the number of 1's is twice the number of 0. What would you use describe this language regular expressions or context free grammar? Justify your answer.

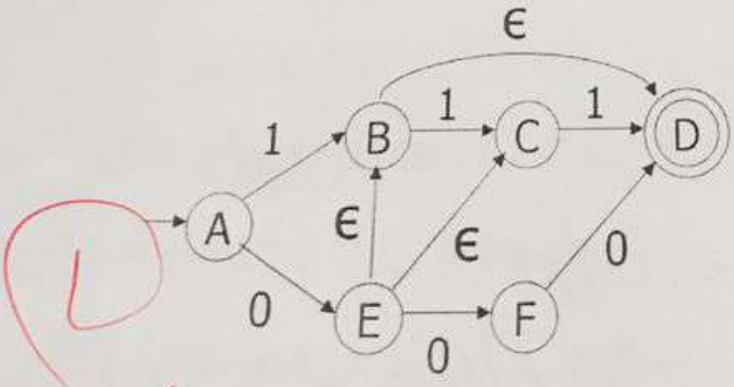
I will use context free grammar.

because it powre full than regular expressions and context free grammar have memory.

Q3) (4 marks) A) Draw an NFA that recognizes the language denoted by the regular expression 1(0+1*).



B) (2 marks) Write a regular expression that describes the language recognized by the following NFA.



{0 0 0}
{1 1 1}
{0 1 1}
{ 0, 1}
{ 1}

(000+111+011+01+1)

0 rejected

C) (2 marks) Represent the above NFA using a table.

	0	1	ε
→A	{F}	{B}	∅
B	∅	{C}	{D}
E	{F}	∅	{B, C}
C	∅	{D}	∅
F	{D}	∅	∅
*D	∅	∅	∅

D) (4 marks) Write an algorithm that uses the above table to determine if a given string is legal.

```
i=0;
Stauts=0;
while (inPut[i]){
if( Stauts==A&& inPut[i]==0){
```