CENG 331 Take-Home Final

Fatih Yıldız
2306793

Q 2.2)

For more accurate analysis, I extended the algorithm to loop through 100 times for each index.

```
for(int i=1; i<100; i++){
        for(int j=1; j<100; j++){
            c[i,j]=0;
            for(int k=1; k<100; k++){
                c[i,j]=c[i,j]+a[i,k]*b[k,j];
            }
        }
}
```

As I proved in the following pages, the misprediction rates are as follows:

1-bit ⇒ 0,019797
2-bit first ⇒ 0,010098
2-bit second ⇒ 0,019998

As we can see, the first 2-bit predicter performed the best, with around 1% misprediction rate. Other two predictors yielded similar results for input size 100. I suppose in bigger input sizes, the 1-bit predicter would outperform the second 2-bit predictor. But the superiority of the first 1-bit predictor would continue.

For the <u>1-bit predictor</u>, I assumed the initial state to be 0 (taken). Here is a summary of the desicions made:

T ⇒ Taken, NT ⇒ Not taken

	Guess	Actual	Result
	T	T	✓
$i=1$			
$j=1$	T	T	✓
$k=1$	T	T	✓
⋮	⋮	⋮	
$k=99$	T	T	✓
$k=100$	T	NT	X
$j=2$	NT	T	X
…			
⋮			
$j=99$	NT	T	X
$j=100$	NT	NT	✓
$i=2$	NT	T	X
$j=1$	T	T	✓
$j=2$	NT	T	X
⋮	⋮		
$j=99$	NT	T	X
$j=100$	NT	NT	✓
⋮			
$i=99$	NT	T	X
$j=1$	T	T	✓
$j=99$	NT	T	X
$j=100$	NT	NT	✓
$i=100$	NT	NT	✓

w_i ⇒ count of wrong desicions on loop i

t_i ⇒ total count of desicions on loop i

$i=1$

$w_i \Rightarrow 99+98$

$t_i \Rightarrow 100.99+100+1$

$i=2\ldots99$

$w_i \Rightarrow 99+98+1$

$t_i \Rightarrow 100.99+100+1$

⋮

$i=100$

$w_i \Rightarrow 0$

$t_i \Rightarrow 1$

Then, the total misprediction rate for input size 100 is:

$$r = \sum_{i=1}^{100} \frac{w_i}{t_i} = \frac{99+98+98.(99+98+1)+0}{100.99+100+1+98(100.99+100+1)+1} = 0{,}019797$$

For the first 2-bit predictor, I assumed the initial state to be 0 (strongly taken). Note that the input size is kept unchanced to observe the effect of predictors on misprediction rate. Here is a summary of the desicions made:

ST → strongly taken T → taken

SNT → strongly not taken NT → not taken

	Guess	Actual	New State
i=1	ST	taken	ST
J=1	ST	taken	ST
k=1	ST	taken	ST
⋮			
k=99	ST	taken	ST
k=100	ST	not taken	T
J=2	T	taken	ST
k=1	ST	taken	ST
⋮			
k=99	ST	taken	ST
k=100	ST	not taken	T
⋮			
J=100	T	not taken	NT
i=100	NT	not taken	SNT

$w_i \Rightarrow$ count of wrong desicions on loop i

$t_i \Rightarrow$ total count of desicions on loop i

i=1

$w_i \Rightarrow 99.1+1$

$t_i \Rightarrow 99.100+100+1$

i=2...99

$w_i = 99.1+1+1$

$t_i = 99.100+100+1$

i=100

$w_i = 0$

$t_i = 1$

Then, the total misprediction rate for input size 100 is:

$$r = \sum_{i=1}^{100} \frac{w_i}{t_i} = \frac{99+1+98(99+1+1)+0}{99.100+100+1+98(99.100+101)+1} = \underline{0.010098}$$

For the <u>second 2-bit predictor</u>, I assumed the inital state to be 00 (not taken). The input size is kept unchanged to observe the effect of predictors on misprediction rate. Here is the summary of the desicions made:

	Guess	Actual	New State
i=1	00(NT)	T	01(NT)
J=1	01(NT)	T	11(T)
k=1	11(T)	T	11(T)
k=2	11(T)	T	11(T)
⋮			
k=99	11(T)	T	11(T)
k=100	11(T)	NT	10(T)
J=2	10(T)	T	11(T)
k=1	11(T)	T	11(T)
⋮			
k=99	11(T)	T	11(T)
k=100	11(T)	NT	10(T)
J=3	10(T)	T	11(T)
⋮			
J=99	10(T)	T	11(T)
J=100	10(T)	NT	00(NT)
i=2	00(NT)	T	01(NT)
⋮			
i=99	00(NT)	T	01(NT)
i=100	00(NT)	NT	00(NT

T → taken
NT → not taken

$w_i \Rightarrow$ count of wrong desicions on loop i

$t_i \Rightarrow$ count of total desicions on loop i

<u>$i = 1 \ldots 99$</u>

$w_i = 99.1 + 1 + 1 + 99$

$t_i = 99.100 + 100 + 1$

<u>$i = 100$</u>

$w_i = 0$

$t_i = 1$

Then, the total misprediction rate for input size 100 is:

$$r = \sum_{i=1}^{100} \frac{w_i}{t_i} = \frac{(99+1+1+99).99 + 0.1}{(99.100+100+1)99 + 1} = \underline{0,019998}$$