

# Advanced Operating System

## Homework 1

學生 : M063040065 鄧子奇

## I. Page replacement implement

主要 implement algorithm 包含:

1. First-In-First-Out
2. Optimal algorithm
3. Enhance-second-chance algorithm
4. The algorithm which I think

首先在程式執行畫面中，我預設建立好 Sample 即測試資料，測試資料包含 3 種類型(隨機產生亂數、隨機產生區域性亂數、自行設計測資)，之後提供使用者選擇換頁演算法。



```
E:\GitHub\HW_OS_PageReplacement\c++(2)\cmake-build-debug\c__2_.exe
success Sample_Randon.txt
success Sample_Locality.txt
success Sample_myData.txt
0->FIFO
1->OPT
2->SEC
3->MyReplacement
4->all
Please select algorithm
```

演算法依序為 FIFO、OPT、SEC、MyReplacement 及最後全部的演算法皆執行。

## II. Execution screen:

### 1. 單一個換頁演算法執行

```
success Sample_Randon.txt
success Sample_Locality.txt
success Sample_myData.txt
0->FIFO
1->OPT
2->SEC
3->MyReplacement
4->all
Please select algorithm
0
Please enter FrameSize
10

DataSize 70000
FIFO    Randon        pagefault 67948
FIFO    Locality      pagefault 60817
FIFO    myData        pagefault 27174請按任意鍵繼續 . . .
```

### 2. 全部執行

```
success Sample_Randon.txt
success Sample_Locality.txt
success Sample_myData.txt
0->FIFO
1->OPT
2->SEC
3->MyReplacement
4->all
Please select algorithm
4

DataSize 70000

FrameSize 10

FIFO    Randon        pagefault 67952
OPT     Randon        pagefault 56979
ESC     Randon        pagefault 67958
FIFO    Locality      pagefault 61013
OPT     Locality      pagefault 46374
ESC     Locality      pagefault 61039
FIFO    myData        pagefault 27262
OPT     myData        pagefault 22853
ESC     myData        pagefault 27261

FrameSize 20

FIFO    Randon        pagefault 66037
OPT     Randon        pagefault 50288
ESC     Randon        pagefault 66022
FIFO    Locality      pagefault 53127
OPT     Locality      pagefault 38270
ESC     Locality      pagefault 53135
FIFO    myData        pagefault 26478
OPT     myData        pagefault 20166
ESC     myData        pagefault 26474

FrameSize 30

FIFO    Randon        pagefault 64062
OPT     Randon        pagefault 45378
ESC     Randon        pagefault 64062
FIFO    Locality      pagefault 46802
OPT     Locality      pagefault 33336
ESC     Locality      pagefault 46827
FIFO    myData        pagefault 25671
OPT     myData        pagefault 18236
ESC     myData        pagefault 25664
```

### III. MyReplacement algo

#### A. 設計方法

此方法類似 LRU，LRU 是利用最久未則替換該 page，我的想法則是在過往的 3500 筆資料中比對目前在 Frame 中的 page 出現的次數，替換數字最小，預期希望數字最少的 page 在未來出現的機率低，所以替換數量最少的 page。例如 Frame 大小 5 其中有 10,20,30,40,50 這 5 個 page，在 3500 筆中 10 出現 2000 次，20 出現 600 次，30 出現 400 次，40 出現 1200 次，50 出現 800 次，再次範例中，page 30 將被替換。

#### B. 實驗說明

本實驗利用自行撰寫的 `get_Sample()` 產生所需的 3 個測試資料，依序為 Randon，Locality，myData。

- Randon：每筆資料隨機從 0~350 的範圍內生成，共 70000 筆資料。
- Locality：每一次生成一組個有局域性的若干筆資料，一組資料大小為  $350/6 \sim 350/4$  之間的個數。在隨機產生一筆資料前，會隨機產生兩數最為區域性範圍，大小 0~350，例如隨機產生一組資料大小為  $350/6 < 60 < 350/4$ ，區域性範圍 30~50，所以 60 個數字從 30~50 中生成。

- myData： 每一次隨機生成數字做為重複次數，模擬大量資料

重複使用的情況時，每一資料產生範圍 0~350。

## C. 實驗目的

對 myReplacement 進行測試，評估其效能與優缺點。

## D. 實驗假設

不考量實驗運行時間。

Page fault 時發生 interrupt。

## IV.Result

### A. Sample(txt)

#### 1. Randon

Sample_Randon - 記事本								
檔案(F)	編輯(E)	格式(O)	檢視(V)	說明(H)				
137	278	204	173	214	50	24	242	95
259	193	165	47	324	194	36	125	282
200	56	182	82	262	226	303	87	56
76	82	134	190	59	264	257	242	113
275	73	130	167	75	296	331	205	184
220	204	104	282	241	92	19	34	106
137	124	75	216	112	131	176	157	171
205	217	44	197	189	280	264	91	162
250	281	75	283	140	153	219	162	110
167	337	246	11	147	188	221	221	149
76	59	103	221	16	110	49	268	16
290	298	148	14	336	30	321	241	139
87	46	209	36	214	83	257	64	124
262	332	152	180	260	311	261	149	253
26	1	218	344	143	228	281	312	328
197	281	9	37	134	76	181	111	4
204	198	14	122	262	166	104	347	144
59	272	11	130	193	123	205	186	224
110	89	57	305	25	50	172	173	214
204	39	154	179	89	94	205	263	292
195	290	57	288	9	295	254	256	93
341	195	93	169	44	294	233	38	34

#### 2. Locality

Sample\_Locality - 記事本

檔案(F)	編輯(E)	格式(O)	檢視(V)	說明(H)			
229	169	182	185	167	156	186	223
161	222	166	219	156	224	227	194
189	178	211	162	190	181	201	226
200	222	229	164	158	228	196	226
184	204	165	170	207	219	228	213
176	177	194	164	220	163	159	197
182	191	204	203	175	211	180	211
211	175	164	156	170	203	226	213
216	196	228	207	194	201	179	211
230	214	214	96	93	80	68	102
84	98	105	63	89	107	95	75
73	95	101	80	74	119	109	93
65	83	83	80	91	103	102	66
120	119	118	69	89	62	91	62
64	66	65	119	75	105	80	89
99	118	77	117	62	62	77	66
86	110	116	82	113	69	80	108
293	292	230	257	279	254	262	235

### 3. myData

Sample\_myData - 記事本

檔案(F)	編輯(E)	格式(O)	檢視(V)	說明(H)			
10	10	303	303	290	290	290	290
206	234	149	149	149	149	14	83
335	233	16	16	69	69	69	37
37	37	37	322	322	322	322	231
231	231	231	341	235	235	235	270
270	306	306	306	306	280	280	280
280	37	37	217	40	40	40	40
279	279	70	70	70	45	45	45
45	99	99	99	99	202	199	199
199	228	228	228	228	46	254	254
254	43	127	127	127	340	64	64
64	64	86	86	86	86	253	200
257	257	257	200	49	49	107	337
337	337	110	110	110	108	108	223
223	223	223	345	345	195	195	289
314	178	253	253	253	253	87	87
331	331	5	38	38	38	38	170
170	170	133	72	72	72	72	10

B. (為方便觀看 result，暫時以 frame 大小為 5)

格式皆為 Index | input | frame 1~frame 5 | swap out

#### 1. FIFO

### i. Randon

69991	312	42	100	248	312	262	Swap out : 60
69992	311	42	100	248	312	311	Swap out : 262
69993	12	12	100	248	312	311	Swap out : 42
69994	109	12	109	248	312	311	Swap out : 100
69995	155	12	109	155	312	311	Swap out : 248
69996	134	12	109	155	134	311	Swap out : 312
69997	147	12	109	155	134	147	Swap out : 311
69998	217	217	109	155	134	147	Swap out : 12
69999	337	217	337	155	134	147	Swap out : 109

pagefault : 69017      interrupt : 69017

### ii. Locality

69991	38	41	79	38	40	34	Swap out : 52
69992	37	41	79	38	37	34	Swap out : 40
69993	44	41	79	38	37	44	Swap out : 34
69994	51	51	79	38	37	44	Swap out : 41
69995	83	51	83	38	37	44	Swap out : 79
69996	78	51	83	78	37	44	Swap out : 38
69997	77	51	83	78	77	44	Swap out : 37
69998	38	51	83	78	77	38	Swap out : 44
69999	70	70	83	78	77	38	Swap out : 51

pagefault : 65256      interrupt : 65256

### iii. myData

69991	253						
69992	253						
69993	101	47	124	85	253	101	Swap out : 44
69994	101						
69995	101						
69996	101						
69997	84	84	124	85	253	101	Swap out : 47
69998	84						
69999	211	84	211	85	253	101	Swap out : 124

pagefault : 27737      interrupt : 27737

## 2. OPT

### i. Randon

69992	311	344	143	311	226	320	swap out : 312
69993	12	344	143	12	226	320	swap out : 311
69994	109	344	143	109	226	320	swap out : 12
69995	155	344	143	155	226	320	swap out : 109
69996	134	344	143	134	226	320	swap out : 155
69997	147	344	143	147	226	320	swap out : 134
69998	217	344	143	217	226	320	swap out : 147
69999	337	344	143	337	226	320	swap out : 217
pagefault : 61957		interrupt : 61957					

## ii. Locality

69992	37	303	70	38	37	335	swap out : 66
69993	44	303	70	38	44	335	swap out : 37
69994	51	303	70	38	51	335	swap out : 44
69995	83	303	70	38	83	335	swap out : 51
69996	78	303	70	38	78	335	swap out : 83
69997	77	303	70	38	77	335	swap out : 78
69998	38	303	70	38	77	335	in Frame
69999	70	303	70	38	77	335	in Frame
pagefault : 53689		interrupt : 53689					

## iii. myData

69992	253	198	43	253	85	123	in Frame
69993	101	198	43	101	85	123	swap out : 253
69994	101	198	43	101	85	123	in Frame
69995	101	198	43	101	85	123	in Frame
69996	101	198	43	101	85	123	in Frame
69997	84	198	43	84	85	123	swap out : 101
69998	84	198	43	84	85	123	in Frame
69999	211	198	43	211	85	123	swap out : 84
pagefault : 24905		interrupt : 24905					

## 3. ESC

### i. Randon

69990	49	84	130	158	49	41	swap out : 259
69991	242	84	130	158	49	242	swap out : 41
69992	165	165	130	158	49	242	swap out : 84
69993	172	165	172	158	49	242	swap out : 130
69994	316	165	172	316	49	242	swap out : 158
69995	40	165	172	316	40	242	swap out : 49
69996	14	165	172	316	40	14	swap out : 242
69997	135	135	172	316	40	14	swap out : 165
69998	24	135	24	316	40	14	swap out : 172
69999	24						
pagefault : 69028		interrupt : 45967					

### ii. Locality



69992	121	121	114	95	128	103	swap out : 131
69993	112	121	112	95	128	103	swap out : 114
69994	156	121	112	95	156	103	swap out : 128
69995	125	121	112	95	156	125	swap out : 103
69996	146	146	112	95	156	125	swap out : 121
69997	96	146	112	96	156	125	swap out : 95
69998	129	146	112	96	129	125	swap out : 156
69999	131	146	131	96	129	125	swap out : 112
pagefault : 65276			interrupt : 42200				

### iii. myData

69992	72						
69993	45	175	0	72	45	71	swap out : 244
69994	118	175	0	72	45	118	swap out : 71
69995	118						
69996	111	111	0	72	45	118	swap out : 175
69997	111						
69998	111						
69999	70	111	70	72	45	118	swap out : 0
pagefault : 27675			interrupt : 14204				

## 4. MyReplacement

### i. Randon

69991	198	24	216	128	110	198	swap out : 250
69992	64	24	216	128	110	64	swap out : 198
69993	12	24	216	128	110	12	swap out : 64
69994	67	24	216	128	110	67	swap out : 12
69995	221	24	216	128	110	221	swap out : 67
69996	342	24	216	128	110	342	swap out : 221
69997	83	24	216	128	110	83	swap out : 342
69998	42	24	216	128	110	42	swap out : 83
69999	104	24	216	128	110	104	swap out : 42
pagefault : 69019			interrupt : 69019				

### ii. Locality

69989	62						
69990	94	216	94	100	128	62	swap out : 71
69991	85	216	85	100	128	62	swap out : 94
69992	73	216	73	100	128	62	swap out : 85
69993	69	216	69	100	128	62	swap out : 73
69994	85	216	85	100	128	62	swap out : 69
69995	46	216	46	100	128	62	swap out : 85
69996	104	216	104	100	128	62	swap out : 46
69997	97	216	97	100	128	62	swap out : 104
69998	100						
69999	77	216	77	100	128	62	swap out : 97
pagefault : 67833			interrupt : 67833				

### iii. myData

69994	56	24	100	62	57	56	swap out : 122
69995	56						
69996	56						
69997	25	24	100	62	57	25	swap out : 56
69998	25						
69999	279	24	100	62	57	279	swap out : 25
pagefault : 27483			interrupt : 27483				

---

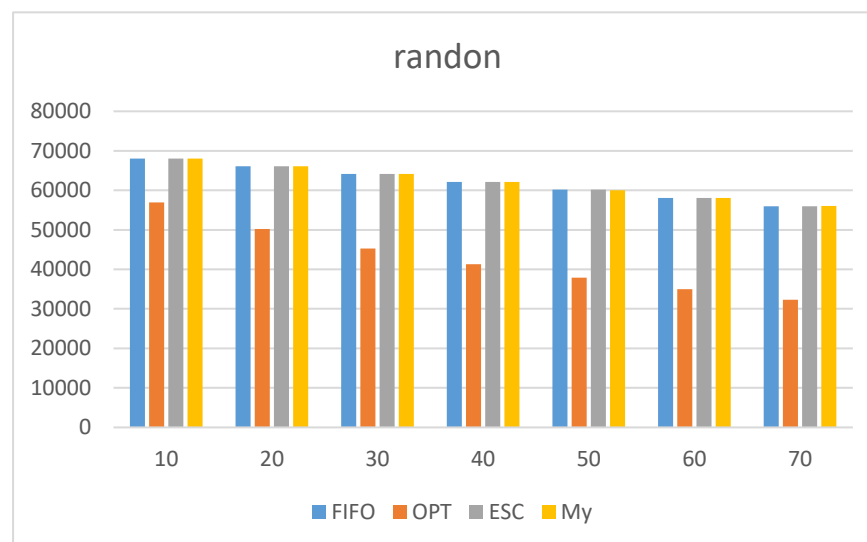
## V. The comparison

本章節分析 FIFO(First In First Out)、OPT(Optimal)、ESC(Enhance

second chance)、myReplacement 之效能

### A. Pagefault 比較:

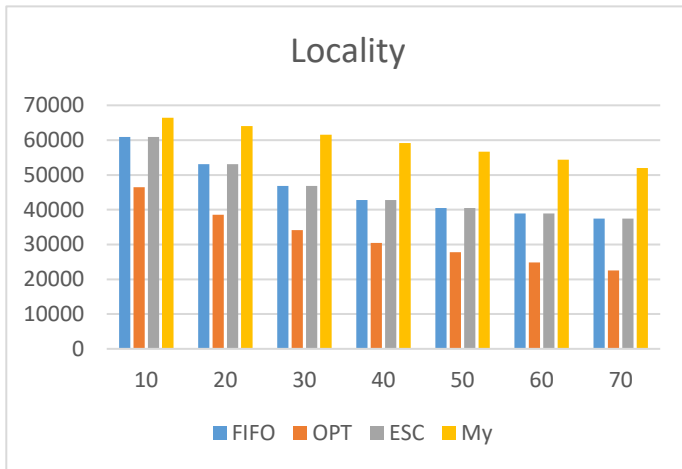
#### 1. Randon



	10	20	30	40	50	60	70
FIFO	27099	26292	25479	24700	23908	23117	22370
OPT	22764	20038	18118	16560	15202	14050	13026
ESC	27099	26292	25479	24700	23908	23117	22374
My	27101	26293	25475	24704	23891	23128	22327

圖 1 Randon page fault 次數與 Frame size 關係圖

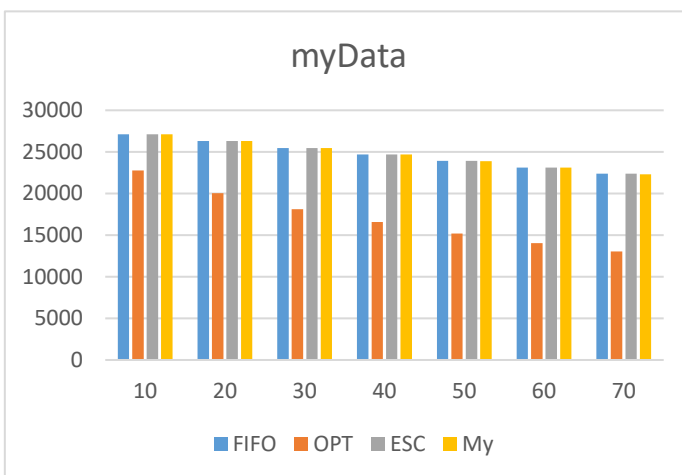
## 2. Locality



	10	20	30	40	50	60	70
FIFO	60932	53060	46863	42815	40474	38872	37442
OPT	46451	38551	34098	30434	27734	24843	22559
ESC	60932	53060	46863	42815	40474	38872	37442
My	66468	64045	61529	59188	56699	54373	51995

圖 2 Locality page fault 次數與 Frame size 關係圖

## 3. myData

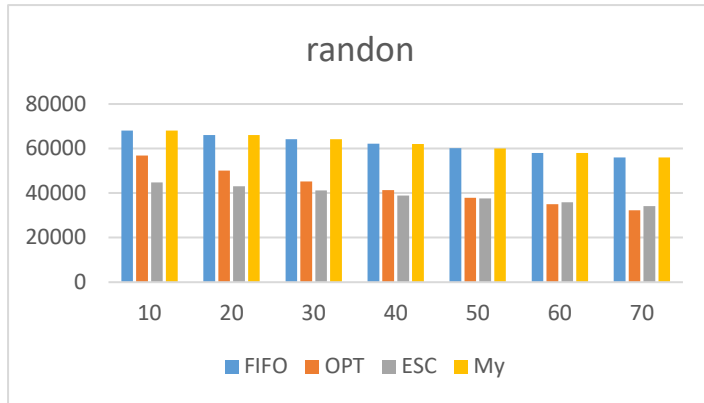


	10	20	30	40	50	60	70
FIFO	27099	26292	25479	24700	23908	23117	22370
OPT	22764	20038	18118	16560	15202	14050	13026
ESC	27099	26292	25479	24700	23908	23117	22374
My	27101	26293	25475	24704	23891	23128	22327

圖 3 myData page fault 次數與 Frame size 關係圖

## B. Interrupt 比較:

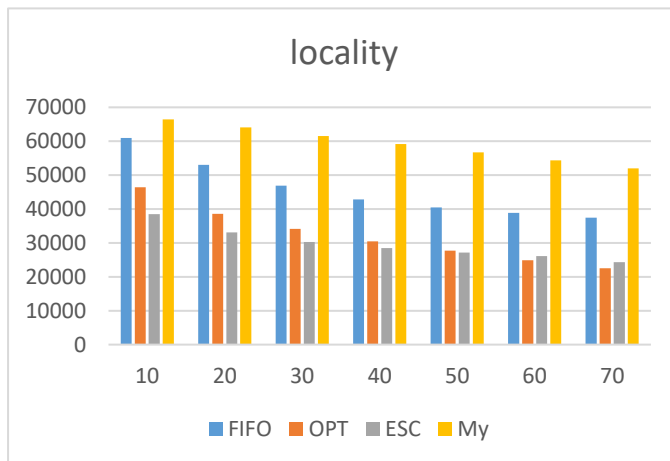
### 1. Randon



	10	20	30	40	50	60	70
FIFO	68064	66124	64162	62133	60156	58046	55994
OPT	56897	50174	45279	41302	37899	34951	32280
ESC	44776	43057	41154	38961	37658	35834	34210
My	68064	66119	64133	62099	60016	58061	56050

圖 4 Randon interrupt 次數與 Frame size 關係圖

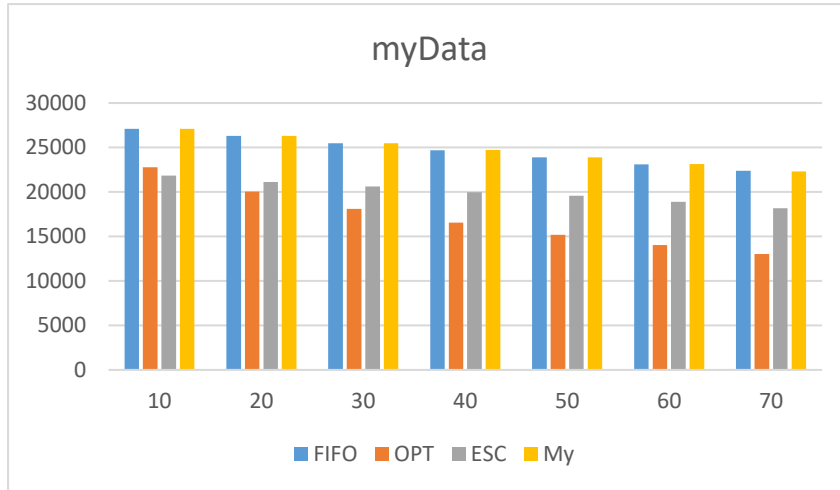
## 2. Locality



	10	20	30	40	50	60	70
FIFO	60932	53060	46863	42815	40474	38872	37442
OPT	46451	38551	34098	30434	27734	24843	22559
ESC	38448	33102	30302	28444	27158	26074	24316
My	66468	64045	61529	59188	56699	54373	51995

圖 5 Locality interrupt 次數與 Frame size 關係圖

## 3. myData



	10	20	30	40	50	60	70
FIFO	27099	26292	25479	24700	23908	23117	22370
OPT	22764	20038	18118	16560	15202	14050	13026
ESC	21858	21115	20636	19974	19591	18884	18170
My	27101	26293	25475	24704	23891	23128	22327

圖 6 myData interrupt 次數與 Frame size 關係圖

## VI. Conclusion

在實驗的過程中，我發現以預測的方式去選擇 **page** 替換並不是一個很好的方法，在數據中不管以哪一種測資為例，實驗結果皆只有比 **FIFO** 好一點，且當 **Frame Size** 變大時，下降速度等於

**FIFO**，但各版本的下降幅度明顯優於 **myReplacement algo**。

通過實驗數據，得到 **myReplacement** 效果等同 **FIFO**，不過卻能險勝 **FIFO**。

在實驗最後必須聲明，因為實驗以不考慮時間為前提去評估各版本 **pagefault** 和 **interrupt** 的差異，若將時間列入考慮，

**myReplacement** 已無任何參考價值，過多使用系統資源，在時間運行上與 **FIFO** 相比十分劣勢。