

# PORTOFOLIO LAPORAN AKHIR

## Mata Kuliah Dasar sistem computer

### Judul Proyek

Sistem Antrian Prioritas Berbasis Assembly

### Identitas

- **Nama** : Muhammmad Rosyhan Anwar
- **NIM** : 2500018111
- **Kelas** : C
- **Mata Kuliah** : Dasar Sistem Komputer

### Deskripsi Singkat Proyek

Proyek ini merupakan aplikasi Sistem Antrian Prioritas berbasis teks yang dibuat menggunakan bahasa pemrograman Assembly dan dijalankan pada emulator EMU8086. Aplikasi dirancang untuk mensimulasikan sistem antrean dengan dua jenis layanan, yaitu Reguler dan Prioritas, di mana antrean prioritas akan dipanggil lebih dahulu.

### Fitur Aplikasi

1. Mengambil nomor antrean **Reguler**
2. Mengambil nomor antrean **Prioritas**
3. Memanggil antrean (mendahulukan prioritas)
4. Keluar dari aplikasi

### Alur Program

**START → Menu Utama → Ambil Reguler / Ambil Prioritas → Kembali ke Menu → Panggil Antrian → Prioritas → Reguler → Menu → Keluar → END**

### Struktur Program

#### Data Segment

- Variabel nomor antrean reguler dan prioritas
- Pesan menu dan output teks

## Code Segment

- Menu utama
- Proses pengambilan antrean
- Proses pemanggilan antrean
- Keluar program

## Source Code Inti

```
.model small
```

```
.stack 100h
```

```
.data
```

```
menu db 13,10,"=== SISTEM ANTRIAN PRIORITAS ===",13,10
```

```
db "1. Ambil Reguler",13,10
```

```
db "2. Ambil Prioritas",13,10
```

```
db "3. Panggil Antrian",13,10
```

```
db "4. Keluar",13,10
```

```
db "Pilih: $"
```

```
msgR db 13,10,"Nomor Reguler: R$"
```

```
msgP db 13,10,"Nomor Prioritas: P$"
```

```
msgCall db 13,10,"Dipanggil: $"
```

```
msgEmpty db 13,10,"Antrian kosong.$"
```

```
regNext db 1
```

```
prioNext db 1
```

```
regCall db 1
```

```
prioCall db 1
```

```
.code
```

```
main proc
```

```
mov ax,@data
```

```
mov ds,ax
```

```
menu_loop:
```

```
    lea dx,menu
```

```
    mov ah,09h
```

```
    int 21h
```

```
    mov ah,01h
```

```
    int 21h
```

```
    sub al,'0'
```

```
    cmp al,1
```

```
    je ambil_reg
```

```
    cmp al,2
```

```
    je ambil_prio
```

```
    cmp al,3
```

```
    je panggil
```

```
    cmp al,4
```

```
    je keluar
```

```
    jmp menu_loop
```

```
ambil_reg:
```

```
    lea dx,msgR
```

```
    mov ah,09h
```

```
    int 21h
```

```
    mov al,regNext
```

```
    add al,'0'
```

```
    mov dl,al
```

```
    mov ah,02h
```

```
    int 21h
```

```
inc regNext  
jmp menu_loop
```

ambil\_prio:

```
lea dx,msgP  
mov ah,09h  
int 21h
```

```
mov al,prioNext  
add al,'0'  
mov dl,al  
mov ah,02h  
int 21h
```

```
inc prioNext  
jmp menu_loop
```

; ===== PANGGIL ANTRIAN (FIXED) =====

panggil:

```
mov al, prioCall    ; AL = nomor prioritas yang akan dipanggil  
cmp al, prioNext  
jb call_prio        ; jika masih ada prioritas
```

```
mov al, regCall  
cmp al, regNext  
jb call_reg         ; jika masih ada reguler
```

```
lea dx,msgEmpty  
mov ah,09h  
int 21h  
jmp menu_loop
```

call\_prio:

lea dx,msgCall

mov ah,09h

int 21h

mov dl,'P'

mov ah,02h

int 21h

mov al,prioCall

add al,'0'

mov dl,al

mov ah,02h

int 21h

inc prioCall

jmp menu\_loop

call\_reg:

lea dx,msgCall

mov ah,09h

int 21h

mov dl,'R'

mov ah,02h

int 21h

mov al,regCall

add al,'0'

mov dl,al

mov ah,02h

int 21h

```
inc regCall  
jmp menu_loop
```

keluar:

```
mov ah,4Ch  
int 21h
```

```
main endp
```

```
end main
```

```
.model small
```

```
.stack 100h
```

```
.data
```

```
menu db 13,10,"=== SISTEM ANTRIAN PRIORITAS ===",13,10
```

```
db "1. Ambil Reguler",13,10
```

```
db "2. Ambil Prioritas",13,10
```

```
db "3. Panggil Antrian",13,10
```

```
db "4. Keluar",13,10
```

```
db "Pilih: $"
```

```
msgR db 13,10,"Nomor Reguler: R$"
```

```
msgP db 13,10,"Nomor Prioritas: P$"
```

```
msgCall db 13,10,"Dipanggil: $"
```

```
msgEmpty db 13,10,"Antrian kosong.$"
```

```
regNext db 1
```

```
prioNext db 1
```

```
regCall db 1
```

```
prioCall db 1
```

```
.code
```

main proc

mov ax,@data

mov ds,ax

menu\_loop:

lea dx,menu

mov ah,09h

int 21h

mov ah,01h

int 21h

sub al,'0'

cmp al,1

je ambil\_reg

cmp al,2

je ambil\_prio

cmp al,3

je panggil

cmp al,4

je keluar

jmp menu\_loop

ambil\_reg:

lea dx,msgR

mov ah,09h

int 21h

mov al,regNext

add al,'0'

mov dl,al

mov ah,02h

int 21h

inc regNext

jmp menu\_loop

ambil\_prio:

lea dx,msgP

mov ah,09h

int 21h

mov al,prioNext

add al,'0'

mov dl,al

mov ah,02h

int 21h

inc prioNext

jmp menu\_loop

; ===== PANGGIL ANTRIAN (FIXED) =====

panggil:

mov al, prioCall ; AL = nomor prioritas yang akan dipanggil

cmp al, prioNext

jb call\_prio ; jika masih ada prioritas

mov al, regCall

cmp al, regNext

jb call\_reg ; jika masih ada reguler

lea dx,msgEmpty

mov ah,09h

int 21h



```
jmp menu_loop
```

```
call_prio:
```

```
    lea dx,msgCall
```

```
    mov ah,09h
```

```
    int 21h
```

```
    mov dl,'P'
```

```
    mov ah,02h
```

```
    int 21h
```

```
    mov al,prioCall
```

```
    add al,'0'
```

```
    mov dl,al
```

```
    mov ah,02h
```

```
    int 21h
```

```
    inc prioCall
```

```
    jmp menu_loop
```

```
call_reg:
```

```
    lea dx,msgCall
```

```
    mov ah,09h
```

```
    int 21h
```

```
    mov dl,'R'
```

```
    mov ah,02h
```

```
    int 21h
```

```
    mov al,regCall
```

```
    add al,'0'
```

```
    mov dl,al
```

```
mov ah,02h
```

```
int 21h
```

```
inc regCall
```

```
jmp menu_loop
```

keluar:

```
mov ah,4Ch
```

```
int 21h
```

```
main endp
```

```
end main
```

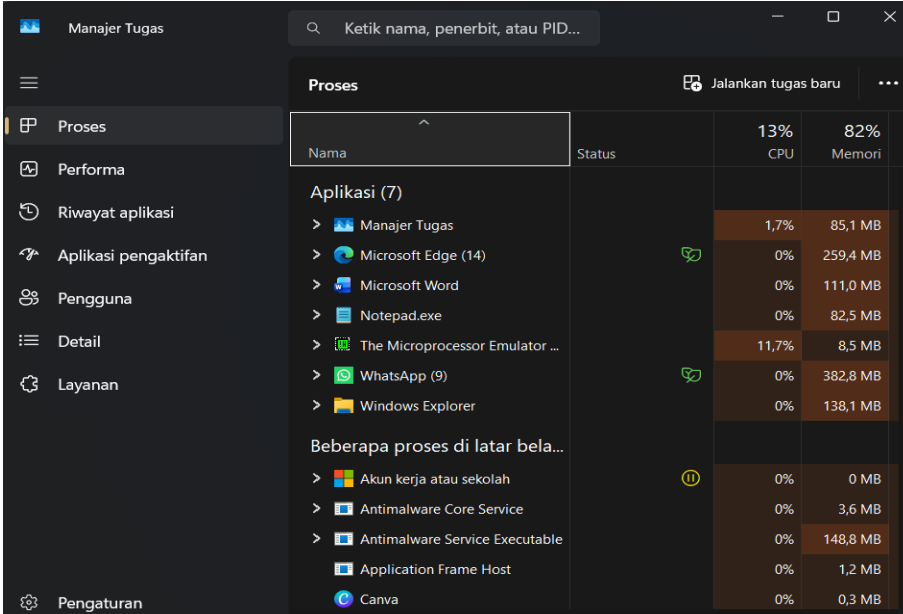
## HASIL ANALISIS KINERJA APLIKASI

### 1. Tujuan Pengujian

Pengujian dilakukan untuk mengetahui penggunaan sumber daya komputer saat aplikasi dijalankan menggunakan Task Manager dan tool monitoring sistem.

### 2. Analisis Task Manager

Tampilan Task Manager saat aplikasi Sistem Antrian Prioritas dijalankan menggunakan EMU 8086.



The screenshot shows the Windows Task Manager application. The 'Proses' (Processes) tab is selected, displaying a list of running applications. The columns shown are 'Nama' (Name), 'Status', 'CPU', and 'Memori' (Memory). The total CPU usage is 13% and total memory usage is 82%.

Nama	Status	CPU	Memori
<b>Aplikasi (7)</b>			
Manajer Tugas		1,7%	85,1 MB
Microsoft Edge (14)		0%	259,4 MB
Microsoft Word		0%	111,0 MB
Notepad.exe		0%	82,5 MB
The Microprocessor Emulator ...		11,7%	8,5 MB
WhatsApp (9)		0%	382,8 MB
Windows Explorer		0%	138,1 MB
<b>Beberapa proses di latar bela...</b>			
Akun kerja atau sekolah		0%	0 MB
Antimalware Core Service		0%	3,6 MB
Antimalware Service Executable		0%	148,8 MB
Application Frame Host		0%	1,2 MB
Canva		0%	0,3 MB

- Aplikasi dijalankan menggunakan EMU 8086 pada sistem operasi Windows.
- Penggunaan CPU sekitar 11–12% saat aplikasi aktif.
- Penggunaan memori sekitar 8,5 MB RAM.
- Tidak terjadi lonjakan penggunaan CPU maupun memori selama program berjalan.
- Aplikasi berjalan stabil tanpa mempengaruhi proses lain.
- Menunjukkan bahwa aplikasi Assembly x86 ringan dan efisien dalam penggunaan sumber daya.

### 3. Analisis Tool Monitoring

Hasil pemantauan kinerja dan suhu perangkat menggunakan aplikasi HWMonitor.

Sensor	Value	Min	Max
<b>LENOVO</b>			
<b>LENOVO LNVN8161216</b>			
Temperatures			
CPU	28.0 °C	28.0 °C	28.0 °C
Intel PCH	42.0 °C	41.0 °C	43.0 °C
Utilization			
Physical Memory Load	87.0 %	86.0 %	87.0 %
Available Physical Me...	1.0 GB	1.0 GB	1.0 GB
Virtual Memory Load	63.6 %	63.2 %	63.7 %
Available Virtual Mem...	6.5 GB	6.4 GB	6.5 GB
<b>Intel Core i3 1215U</b>			
Voltages			
VID (Max)	0.902 V	0.750 V	0.938 V
IA Offset	+0.000 V	+0.000 V	+0.000 V
GT Offset	+0.000 V	+0.000 V	+0.000 V
Ring Offset	+0.000 V	+0.000 V	+0.000 V
System Agent Offset	+0.000 V	+0.000 V	+0.000 V
L2 E-Core Offset	+0.000 V	+0.000 V	+0.000 V
Temperatures			
P-Cores (Max)	46.0 °C	40.0 °C	50.0 °C
E-Cores (Max)	41.0 °C	39.0 °C	47.0 °C
Package	46.0 °C	41.0 °C	47.0 °C
Powers			
Package	5.61 W	2.92 W	8.46 W
IA Cores	3.90 W	1.59 W	6.78 W
GT	0.01 W	0.00 W	0.16 W
Power Max (PL1)	27.00 W	27.00 W	27.00 W
Short Power Max (PL2)	51.00 W	51.00 W	51.00 W
Max Peak Power (PL4)	123.00 W	123.00 W	123.00 W
Currents			
VR Out	6.59 A	2.82 A	12.24 A
IcMax	80.00 A	80.00 A	80.00 A
Clocks			
P-Cores (Max)	3192.2 MHz	1396.6 MHz	3193.2 MHz
E-Cores (Max)	2394.1 MHz	1496.3 MHz	2394.9 MHz
CPU BCLK	99.8 MHz	99.8 MHz	99.8 MHz
LLC/Ring	2593.7 MHz	788.0 MHz	2594.3 MHz
Memory Controller	532.0 MHz	532.0 MHz	1330.4 MHz
Memory	1064.1 MHz	1064.1 MHz	1596.1 MHz
Utilization			
Processor	27.7 %	24.2 %	61.6 %
P-Cores	55.4 %	48.2 %	121.7 %
E-Cores	0.1 %	0.1 %	1.6 %
<b>WD PC SN740 SDDPMQD-512...</b>			
Temperatures			
Assembly	37.0 °C	37.0 °C	38.0 °C
Sensor 1	50.0 °C	49.0 °C	56.0 °C
Sensor 2	37.0 °C	37.0 °C	38.0 °C
Counters			

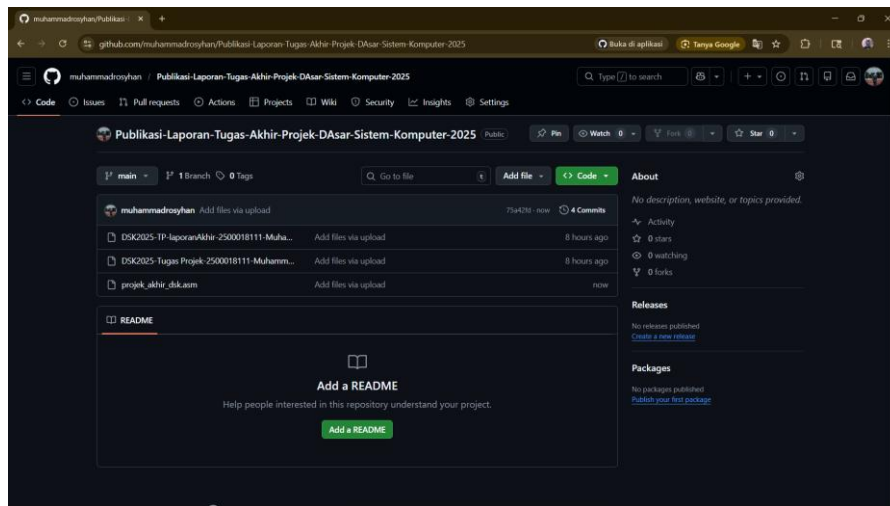
- Pengujian dilakukan saat aplikasi Sistem Antrian Prioritas berjalan di EMU 8086.
- Suhu CPU **stabil** di kisaran **41–46°C**, masih dalam batas normal.
- Daya CPU **rendah**, rata-rata sekitar **5–8 Watt**, menunjukkan beban ringan.
- Utilisasi prosesor sekitar 25–30%, tidak membebani sistem secara berlebihan.
- Memori fisik terpakai tinggi karena aplikasi lain berjalan bersamaan, namun aplikasi tetap stabil.
- Tidak terjadi **overheating** maupun lonjakan daya selama pengujian.
- Aplikasi tergolong **efisien dan aman dijalankan dalam jangka waktu lama**.

#### 4. Kesimpulan Kinerja

Aplikasi Sistem Antrian Prioritas tergolong ringan, efisien, dan stabil, serta tidak membebani sistem komputer. Sangat sesuai sebagai media pembelajaran kinerja komputer berbasis Assembly.

### UNGGAHAN GITHUB & YOUTUBE

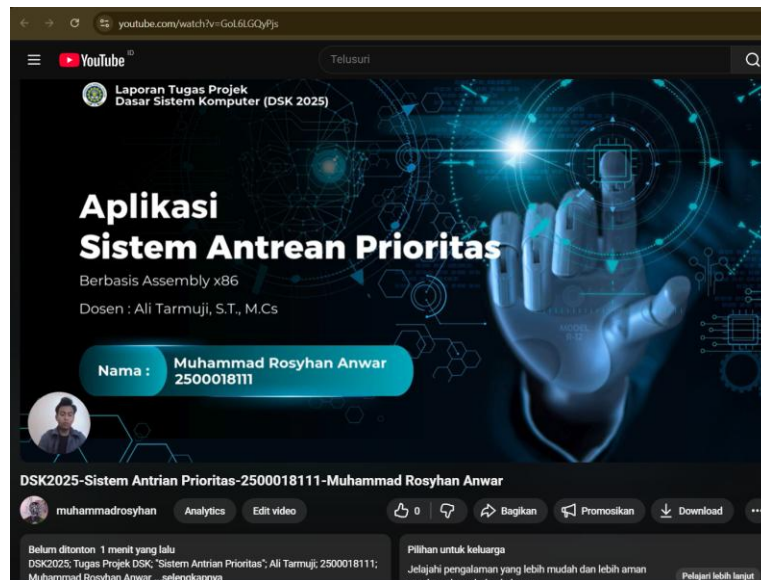
#### GitHub:



Link Github:

<https://github.com/muhammadrosyhan/Publikasi-Laporan-Tugas-Akhir-Projek-DAsar-Sistem-Komputer-2025.git>

**YouTube:**



**Link Youtube:**

<https://youtu.be/GoL6LGQyPjs>

## **ANALISIS Pengerjaan Proyek**

### **1. Tinjauan Waktu**

Proyek diselesaikan sesuai jadwal dengan tahapan perancangan, pengkodean, dan pengujian.

### **2. Ketercapaian Spesifikasi**

Seluruh fitur utama berhasil diimplementasikan dan berjalan sesuai rancangan.

### **3. Biaya**

Tidak memerlukan biaya tambahan karena menggunakan software emulator dan perangkat pribadi.

### **4. Kendala**

- Pemahaman logika Assembly
- Debugging alur pemanggilan antrean

### **5. Tantangan Pengembangan**

Pengembangan lanjutan dapat mencakup:

- Penyimpanan data permanen
- Nomor antrean multi-digit
- Tampilan lebih interaktif

