Assignment 3 Result Report 通訊二 110503531 劉泰君

1.Compile Result

Arithmetic

```
cornertw@DESKTOP-HD1IFP1:/mnt/c/Users/Max Liu/NCU_Documents/arcd$ make
[ 25%] Built target arcd
[ 50%] Built target adaptive_model
Scanning dependencies of target arcd_stream
[ 62%] Building C object examples/CMakeFiles/arcd_stream.dir/arcd_stream.c.o
[ 75%] Linking C executable arcd_stream
[ 75%] Built target arcd_stream
Scanning dependencies of target codec_tests
[ 87%] Building CXX object tests/CMakeFiles/codec_tests.dir/codec_tests.cpp.o
[ 100%] Linking CXX executable codec_tests
[ 100%] Built target codec_tests
```

Huffman

```
cornertw@DESKTOP-HD1IFP1:/mnt/c/Users/Max Liu/NCU_Documents/huffman-main$ make
cc -g -Wall -Werror -Wextra -00 -std=c11 -D_POSIX_C_SOURCE=2 -c -o huffcode.o huffcode.c
cc -o huffcode huffcode.o libhuffman.a
```

2. Execute Result

Arithmetic

- Encoding
- input file is "runarcd.txt",
- encoding file is "outputarcd.txt"

```
| tee outputarcd.txt | tee ou
```

- Decoding
- input file is "outputarcd.txt",
- decoding file is "decodingarcd.txt"

cornerts@DESKTOP-HD1FP1:/mnt/c/Users/Max Liu/NCU_Documents/arcd/examples\$./arcd_stream -d < outputarcd.txt | tee decodingarcd.txt 「隨著科技發展,無人機現在能執行許多任務,但電池更換都需要手動,本專案希望透過自動充電平台讓無人機可以有更長久、更廣闊的飛行時間與範圍,而不是必須在執行任務途中環要返回基地進行電池更換」;據外媒NewAtlas報導,大多數多旋翼無人機只能飛行30分鐘左右,之後電池需充電13/2/時,這大幅限制了它們的實際應用。雖然目前可用人力手動更換電池,但今日若需要一次出動大量的無人機,那耗費的時間和人力資源將相當龐大。另外天氣的狀況也會對電池的持續時間造成影響,大風會造成無人機電機更耗功率的運轉,濕氣也會造成無人機重量加重,導致電池消耗的加速,再來溫度的降低會造成電池中鋰聚合物的活性降低,從而導致無人機更快的失去電力。為了解決此問題,本專案提出一個能夠普及的無人機充電平台概念,不僅能夠進行無人機的充電,也能與後端資料庫系統連接進行資料的分析計算,讓無人機未來能執行多元的任務並提升其調度和執行任務的效率,且將能更有效的運用無人機每一份珍貴的電力,使其發揮最大的價值。執行時間(毫秒): 321

Huffman

- Encoding
- input file is "runhuff.txt", encoding file is "outputhuff.txt"

cornertw@DESKTOP-HD1IFP1:/mnt/c/Users/Max Liu/NCU_Documents/DSA_project/huffmain/huffman-main\$./huffcode -i runhuff.txt -o outputhuff.txt -c 執行時間(毫秒):1090.000000

- Decoding
- input file is "outputhuff.txt", decoding file is "runhuff.txt"

cornertw@DESKTOP-HD1IF91:/mmt/c/Users/Max Liu/NCU_Documents/DSA_project/huffmain/huffman-main\$./huffcode -i outputhuff.txt -o runhuff.txt -d 執行時間(毫秒):877.000000

3. Analysis

Because one of my classmates and I coded with longer data content and the other with shorter data, I found that Huffman coding is faster when processing large data, while arithmetic coding is faster when processing a small amount of data.