

***ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ***  
***ΤΗΜΜΥ***  
***ΠΑΡΑΛΛΗΛΟΙ ΚΑΙ ΔΙΚΤΥΑΚΟΙ***  
***ΥΠΟΛΟΓΙΣΜΟΙ***

***PROJECT 2o***  
***Serial and Parallel FFT***  
***Implementation***

***ΑΛΕΞΑΝΔΡΙΔΗΣ ΓΙΩΡΓΟΣ***  
***ΓΡΕΤΣΙΣΤΑΣ ΒΑΣΙΛΗΣ***  
***ΜΑΡΕΛΑΣ ΓΙΩΡΓΟΣ***

## ΑΣΚΗΣΗ 1 και 2:

Για τις ασκήσεις 1 και 2 επισυνάπτουμε τα αρχεία με τον αντίστοιχο κώδικα της σειριακής υλοποίησης και της παράλληλης.

- 1) serial\_fft.c run: gcc -g serial\_fft.c -o serial\_fft -lm  
2) openmp\_fft.c run: gcc -g -fopenmp openmp.c -o openmp -lm

Τα 2 αρχεία εκτύπωνουν τα αποτελέσματα της εξόδου στο αρχείο 'results.txt'.

## ΑΣΚΗΣΗ 3:

**k** : δύναμη του 2. Καθορίζει το πλήθος των inputs

**m** : Ο αριθμός των threads που χρησιμοποιήθηκαν κατά την εκτέλεση του κώδικα.

Για m=1 δίνονται οι χρόνοι της εκτέλεσης του σειριακού αλγορίθμου

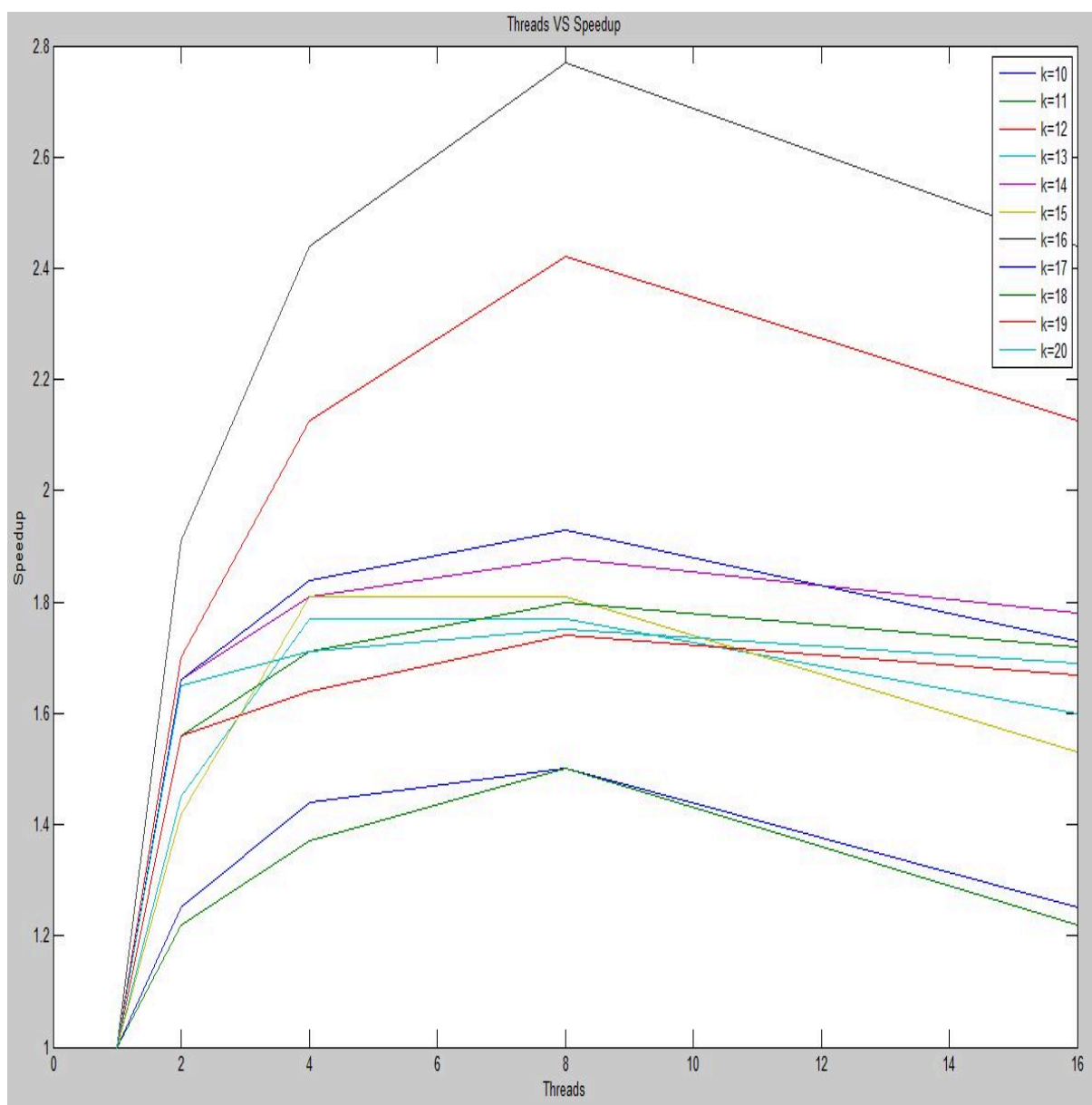
- ✓ Ο κώδικας εκτελέστηκε σε επεξεργαστή 1,4 GHz Intel Core i5 (2 Cores)
- ✓ Η επιλογή των χρόνων είναι αποτέλεσμα του μέσου όρου 5 εκτελέσεων του αντίστοιχου κώδικα. Αποτυπώνονται σε sec.
- ✓ Το Speed up υπολογίζεται από τον τύπο  $S_p = T_s/T_p$ , όπου  $T_s$  ο σειριακός χρόνος εκτέλεσης και  $T_p$  ο παράλληλος.

<i>k vs m</i>	<i>1</i>	<i>2</i>	<i>4</i>	<i>8</i>	<i>16</i>	<i>m&gt;16</i>
<b>10</b>	0.0075	0.006 (1.25)	0.0052 (1.44)	0.0050 (1.5)	0.006 (1.25)	Overhead
<b>11</b>	0.011	0.009 (1.22)	0.0080 (1.37)	0.0073 (1.5)	0.009 (1.22)	Overhead
<b>12</b>	0.034	0.020 (1.7)	0.016 (2.125)	0.014 (2.42)	0.016 (2.125)	Overhead
<b>13</b>	0.048	0.033 (1.45)	0.027 (1.77)	0.027 (1.77)	0.030 (1.6)	Overhead
<b>14</b>	0.10	0.060 (1.66)	0.055 (1.81)	0.053 (1.88)	0.056 (1.78)	Overhead
<b>15</b>	0.20	0.14 (1.42)	0.11 (1.81)	0.11 (1.81)	0.13 (1.53)	Overhead
<b>16</b>	0.61	0.32 (1.91)	0.25 (2.44)	0.22 (2.77)	0.25 (2.44)	Overhead
<b>17</b>	0.83	0.50 (1.66)	0.45 (1.84)	0.43 (1.93)	0.48 (1.73)	Overhead
<b>18</b>	1.71	1.10 (1.56)	1.00 (1.71)	0.95 (1.8)	0.99 (1.72)	Overhead
<b>19</b>	3.44	2.20 (1.56)	2.10 (1.64)	1.98 (1.74)	2.05 (1.67)	Overhead
<b>20</b>	7.00	4.22 (1.65)	4.09 (1.71)	4.00 (1.75)	4.12 (1.69)	Overhead

Πίνακας 1.1 'Χρόνοι εκτέσης συναρτήσεων του Πλήθους Εισόδου – Αριθμός Threads '

Στις εκτελέσεις του παράλληλου κώδικα για  $m \geq 32$  όλοι οι χρόνοι μας ήταν ίσοι και μεγαλύτεροι από τον παράλληλο χρόνο εκτέλεσης για  $m = 2$ . Δεν αναφέρονται παραπάνω γιατί θεωρήσαμε ορατό το overhead, δεδομένου πως ξεπερνούσαν τον χρόνο παράλληλης εκτέλεσης για  $m=2$ . Όπως παρατηρείται και από τον πίνακα ο αριθμός των threads που μας έδωσε το μεγαλύτερο speed up ήταν το  $m = 8$  για όλες τις δυνατές τιμές του  $n$  (δεδομένου του 2-πύρηνου επεξεργαστή στον οποίο δοκιμάσαμε τον κώδικα).

Το ίδιο αποτυπώνεται στο γράφημα παρακάτω, όπου βλέπουμε πως η πιο αργή εκτέλεση είναι η σειριακή για  $m = 1$  και η την καλύτερη χρονοβελτίωση την έχουμε για  $m=8$ . Συγκεκριμένα για είσοδο στοιχείων  $2^{16}$  πετυχαίνουμε το μέγιστο speed up.



Γράφημα 1.1 – ‘Threads vs Speed up’