ΑΛΓΟΡΙΘΜΟΙ ΚΑΙ ΠΡΟΧΩΡΗΜΕΝΕΣ ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

**Διερεύνηση προσεγγίσεων επίλυσης του προβλήματος 0-1 σακιδίου**

**ΤΕΧΝΙΚΗ ΕΚΘΕΣΗ**

**ΤΖΙΑΤΖΟΣ ΓΕΩΡΓΙΟΣ**

**ΑΜ: 73**

**ΤΟ ΠΡΟΒΛΗΜΑ**

Το πρόβλημα 0­1 σακιδίου (0­1 knapsack) αφορά ένα σύνολο από αντικείμενα για τα οποία γνωρίζουμε το βάρος και την αξία κάθε αντικειμένου. Ζητείται η επιλογή ενός υποσυνόλου των αντικειμένων έτσι ώστε το συνολικό βάρος από τα επιλεχθέντα αντικείμενα να μην ξεπερνά μια συγκεκριμένη τιμή βάρους και ταυτόχρονα να επιτυγχάνεται η μεγαλύτερη δυνατή αξία. Το πρόθεμα 0­1 στο όνομα του προβλήματος υποδηλώνει ότι κάθε αντικείμενο μπορεί είτε να επιλεχθεί είτε να μην επιλεχθεί στο σύνολό του και όχι τμηματικά.

**ΠΡΟΣΕΓΓΙΣΕΙΣ ΕΠΙΛΥΣΗΣ**

Αρχικά δημιουργούμε τα στιγμιότυπα χρησιμοποιώντας τον generator του άρθρου [Pis99]και τον κώδικα που βρίσκεται στη διεύθυνση (<http://hjemmesider.diku.dk/~pisinger/generator.c>). Δημιουργούμε από 5 στιγμιότυπα για κάθε συνδυασμό των ακόλουθων παραμέτρων: n={10,50,100,500}, r={50,100,500,1000} και type={1,2,3,4}, δηλαδή σύνολο 5 × 4 × 4 × 4 = 320 στιγμιότυπα. Για να παράγουμε αυτά τα στιγμιότυπα φτιάχνουμε ένα shell script.

Η επίλυση του προβλήματος έγινε με χρήση γλώσσας προγραμματισμού c++ και υλοποιήθηκαν για κάθε περίπτωση διαφορετικές συναρτήσεις.

Οι αλγόριθμοι επίλυσης είναι οι παρακάτω:

**Άπληστη μέθοδος**

Η μέθοδος αυτή υπολογίζει για κάθε αντικείμενο το λόγο αξία προς βάρος και δίνει προτεραιότητα στην εισαγωγή των αντικειμένων με τις μεγαλύτερες τιμές και μέχρι να μην μπορεί να προστεθεί άλλο αντικείμενο στο σακίδιο.

**Εξαντλητική απαρίθμηση συνδυασμών**

Η μέθοδος αυτή δοκιμάζει όλους τους πιθανούς συνδυασμούς τοποθέτησης αντικειμένων στο σακίδιο (ανά ένα αντικείμενο, ανά δύο αντικείμενα, ανά τρία αντικείμενα κ.ο.κ.) και επιλέγεται η πλέον συμφέρουσα. Η μέθοδος αυτή δίνει ακριβή αποτελέσματα, αλλά λόγο του μεγάλου αριθμού των συνδυασμών κρίνεται χρονοβόρα.

**Διακλάδωση και φραγή** Η μέθοδος αυτή αποτελεί βελτίωση της μεθόδου πλήρους απαρίθμησης. Αρχικά τα αντικείμενα διατάσσονται σε φθίνουσα σειρά αξίας προς βάρος. Στη συνέχεια δημιουργείται ένα δυαδικό δένδρο μερικών λύσεων. Σε κάθε κόμβο του δένδρου λαμβάνεται η απόφαση επιλογής ενός αντικειμένου η οποία οδηγεί στο αριστερό υποδένδρο ενώ η μη επιλογή του οδηγεί στο δεξιό υποδένδρο. Επιπλέον σε κάθε κόμβο καταγράφεται το συνολικό βάρος και η συνολική αξία των αντικειμένων του.

**Δυναμικός προγραμματισμός**Στην μέθοδος αυτή υλοποιείται κώδικας που χρησιμοποιεί αναδρομή, με επίλυση μικρότερων προβλημάτων και καταγραφή ενδιάμεσων αποτελεσμάτων σε έναν πίνακα ώστε να οδηγηθούμε σε λύση του προβλήματος.

**Εξειδικευμένος επιλυτής**

Η μέθοδος επίλυσης με αυτό τον τρόπο κάνει χρήση της δυνατότητας που παρέχει η google με την βοήθεια του λογισμικού OR-tools και εξειδικευμένων επιλυτών για το πρόβλημα του του 0/1 σακιδίου.

Για όλες τις παραπάνω περιπτώσεις αναπτύχθηκε κώδικας που επιστρέφει την αξία και τον χρόνο που χρειάστηκε για να παραχθεί το αποτέλεσμα. Για τις επίλυση του προβλήματος αναπτύχθηκαν 2 κώδικες, ένας που προσφέρει την λύση για τις 4 πρώτες περιπτώσεις και ένας δεύτερος για την τελευταία περίπτωση, όπου εκτελείτε με την βοήθεια του or-tools.

Τα αποτελέσματα λαμβάνονται σε csv αρχεία όπου με την βοήθεια βιβλίου εργασίας του excel λαμβάνουμε συγκριτικά γραφήματα για όλες τις περιπτώσεις. Προκειμένου να μπορέσουμε να εξάγουμε πιο ασφαλή συμπεράσματα , από την μελέτη των γραφημάτων, τα ομαδοποιήσαμε ανάλογα με των αριθμό των αντικειμένων που διαθέτουμε σε κάθε στιγμιότυπο. Έτσι τα αρχικά 320 στιγμιότυπα χωρίστηκαν σε 4 γραφήματα όπου το καθένα περιλαμβάνει από 80 στιγμιότυπα.

**Αποτελέσματα**

Το πρώτο γράφημα που δημιουργήσαμε περιελάβανε συνολικά 10 αντικείμενα. Από την μελέτη αυτού του γραφήματος βλέπουμε ότι όλες οι μέθοδοι παράγουν αποτελέσματα πολύ κοντά το ένα στο άλλο , πράγμα που μας οδηγεί στο συμπέρασμα ότι αποδίδουν ικανοποιητικά και οι 5 μέθοδοι για μικρό αριθμό αντικειμένων. Αν θέλαμε να απομονώσουμε μια μέθοδο, αυτή θα ήταν μέθοδο της Εξαντλητική απαρίθμηση συνδυασμών, που λόγο του μικρού αριθμού των αντικειμένων δεν κάνει απαγορευτική την εξέταση όλων των δυνατών συνδυασμών και παράγει το ακριβέστερο αποτέλεσμα.

Επίσης παρουσιάζω το γράφημα για τα 50 αντικείμενα . Συνολικά και σε αυτό το γράφημα φαίνονται 80 στιγμιότυπα. Από το γράφημα αυτό βλέπουμε ότι η μέθοδος με τα καλύτερα αποτελέσματα είναι η μέθοδος του δυναμικού προγραμματισμού τόσο με απευθείας εκτέλεση του κώδικα όσο με χρήση του επιλυτή or-tools. H άπληστη μέθοδος και η μέθοδος της Διακλάδωση και φραγή δίνει παρόμοια αποτελέσματα με ελαφρώς ταχύτερη τη μέθοδος της Διακλάδωσης και φραγής.

Εν συνεχεία παραθέτουμε αντίστοιχο γράφημα για 100 αντικείμενα. Παρατηρούμε ότι η μέθοδος της Εξαντλητική απαρίθμησης συνδυασμών δεν μπορεί να αποδώσει γιατί για να βγάλει αποτέλεσμα απαιτεί να κάνει 2100 συγκρίσεις. Οπότε πρακτικά είναι ανέφικτο να κάνει όλες τις δυνατές συγκρίσεις και να δώσει την σωστή λύση. Και σε αυτή την περίπτωση ο or-tools επιλυτής δίνει τα καλύτερα αποτελέσματα. Τα αποτελέσματα με την άπληστη μέθοδο και την Διακλάδωση και φραγή κρίνονται ικανοποιητικά αποδίδοντας μάλιστα αποτέλεσμα σε πολύ σύντομο χρόνο.

Το τελευταίο γράφημα που παρουσιάζω είναι για τα 500 αντικείμενα. Και σε αυτήν την περίπτωση η μέθοδος της εξαντλητικής απαρίθμησης δεν μπορεί να αποδώσει μιας και απαιτούνται 2500 συγκρίσεις. Η μέθοδος με την καλύτερη απόδοση με βάση το χρόνο που απαιτείται για την πραγματοποίηση της είναι η μέθοδος που κάνει χρήση τον επιλυτή του or-tools.

**Συμπεράσματα**

Στην παρούσα εργασία μελετήθηκε το πρόβλημα του 01 σακιδίου με χρήση διαφορετικών μεθόδων επίλυσης του. Από τις μεθόδους που χρησιμοποιήσαμε παρατηρούμε ότι αυτή που μας παρέχει ακριβή αποτελέσματα είναι η μέθοδος της εξαντλητικής απαρίθμησης. Όμως για μεγάλο αριθμό αντικειμένων δεν είναι εφικτό να πραγματοποιηθούν όλες οι δυνατές συγκρίσεις και να βρεθούνε τα αντικείμενα που παρέχουν την μεγαλύτερη δυνατή αξία στην χωρητικότητα που μας δίνεται. Το πρόβλημα αυτό αρχίζει να παρουσιάζεται ακόμα από τα 50 αντικείμενα όπου απαιτούνται 250 συγκρίσεις. Για τον λόγο αυτό καταφεύγουμε σε προσεγγιστικές μεθόδους. Η άπληστη μέθοδος είναι η πιο γρήγορη από όλες, αλλά παρουσιάζει προβλήματα όταν τα αντικείμενα έχουν μεγάλη διαφορά στον λόγο ή ο λόγος είναι ίσος. Η μέθοδος της διακλάδωσης και φραγής παρουσιάζει ανάλογα προβλήματα με την άπληστη μέθοδο. Η μέθοδος του δυναμικού προγραμματισμού είναι καλύτερη από τις προαναφερθείσες , όπου με χρήση πολυδιάστατων πινάκων ελέγχει ένα μεγάλο αριθμό συνδυασμών από αντικείμενα που θα μπορούνε να χρησιμοποιηθούν για να δώσουν λύση στο πρόβλημα. Ο χρόνος που απαιτείται για να βρεθεί η λύση κρίνεται ικανοποιητικός. Η μελέτη του προβλήματος με την βοήθεια του or-tools επιλυτή μπορεί να θεωρηθεί η καλύτερη δυνατή, η μέθοδος αυτή είναι βελτίωση του δυναμικού προγραμματισμού και αποδίδει καλύτερα αποτελέσματα από αυτόν και μάλιστα σε μικρότερο χρόνο.

**Αποθετήριο**

Όλοι οι κώδικες μαζί με τας .csv αρχεία καθώς και το script με την παραγωγή των 320 στιγμιότυπων βρίσκονται αναρτημένα στο αποθετήριο [**https://github.com/tziatzos/knapsack**](https://github.com/tziatzos/knapsack).

**Ιστοσελίδα**

Στην ιστοσελίδα <https://tziatzos.github.io/> παρουσιάζονται επιγραμματικά τα αποτελέσματα της εργασίας και συμπεράσματα που προκύπτουν από αυτά.