

Programming Assignment 2: DNS Client

על מנת לממש את dnsQuery באופן הנדרש השתמשנו בשלושה משתנים גלובליים:

```
SOCKET DNS_sock;
SOCKADDR_IN DNS_sock_addrin;
unsigned short query_ID = 0;
```

SOCKET ו IN_SOCKADDR - עבור תקשורת עם שרת ה DNS.
Query_ID - מזהה עבור חבילת ה DNS הנשלחת.

מבני הנתונים בתכנית:

- (1) Struct מסוג DNS_header, מכיל את שדות ה header של הודעת ה DNS.
- (2) Struct מסוג DNS_qinfo, מכיל את שדות ה question של הודעת ה DNS ללא השדה qname אשר מוגדר באמצעות מצביע נפרד.
- (3) Struct מסוג DNS_rinfo, מכיל את שדות ה Resource של הודעת ה DNS ללא השדה rdata אשר מוגדר באמצעות מצביע נפרד.

על מנת ששדות המבנים לא ירופדו בזיכרון באפסים, כדי שגודלם (sizeof) יהיה זהה לגודלם בהודעת ה DNS, נשעה שימוש בפקודה:

```
#pragma pack(1)
```

פונקציית ה - main:

ראשית, הפונקציה בודקת את שהתקבל לפחות ארגומנט אחד מה CMD ושהארגומנט הראשון הוא כתובת IP בפורמט תקין. במידה ולא, תודפס הודעה שגיאה וריצת התכנית תיעצר.

הפונקציה מאתחלת את הסוקט הגלובלי, מבקשת מהמשתמש להזין domain name, בודקת שהוא בפורמט תקין וקוראת לפונקציה dnsQuery, אם התקבלה תשובה, מדפיסה את ה IP שהתקבל. אם המשתמש הזין domain name בפורמט לא תקין או שלא מתקבלת תשובה משרת ה DNS, יודפסו הודעות שגיאה מתאימות. הפונקציה תמשיך בפעולה זו עד שהמשתמש יזין quit ויסיים את ריצת התכנית.

פונקציית dnsQuery:

הפונקציה העיקרית בתכנית, מתקשרת עם שרת ה DNS.

שליחה: הפונקציה בונה את ה structs הרלוונטיים, ממירה את שם ה domain לפורמט המתאים וממלאת את הבאפר ב structs ובשם ה domain באמצעות שימוש במצביעים. הפונקציה שולחת דרך הסוקט ב-UDP את הבאפר שנבנה אל שרת ה DNS.

קבלה: הפונקציה מקבלת תשובה משרת ה DNS אל הבאפר דרך סוקט ה UDP. הפונקציה בונה מהבאפר המתקבל את structs הרלוונטיים, בודקת בשדה ה ancourt של ה header אם התקבלו תשובות. אם התקבלו תשובות הפונקציה מחפשת את התשובה האחרונה ואוספת ממנה את כתובת ה IP שהתקבלה.

הפונקציה מחזירה מצביע מסוג HOSTENT, אם לא התקבלה תשובה יוחזר מצביע ל NULL ותודפס הודעת שגיאה מתאימה. אם התקבלה תשובה כתובת ה IP שהתקבלה נכנסת לשדה ה h_addr_list של ה- HOSTENT המוחזר.

הפרמטרים בהודעת ה query לשרת ה DNS הם:

Header: RD = 1 → Recursion Desired

Question: qtype = 1 → A → Host Address, qclass = 1 → IN → The Internet

פרמטר ה ID של ה header מאותחל לאפס וגדל בכל בקשה לשרת ה DNS.

כאשר כל שאר השדות מאופסים.

כאשר מתקבלת תשובה משרת ה DNS, נבדוק כמה תשובות התקבלו באמצעות שדה ה ANCOUNT של ה header וניקח את כתובת ה IP מהתשובה האחרונה של שהתקבלה, כתובת זו תמצא בשדה RDATA של התשובה האחרונה.

פונקציית get_domain_name:

מקבלת שם domain מהמשתמש, מסירה רווחים מיותרים מתחילת וסוף המחרוזת. מחזירה 0 אם המשתמש הזין quit, אחרת מחזירה ערך שונה מאפס.

פונקציית remove_leading_spaces:

מסירה רווחים מתחילת המחרוזת, אם יש כאלה.

פונקציית check_domain_name:

בודקת אם מחרוזת היא בפורמט תקין של domain name וממירה תווים ל lowercase:

- אורך: לפחות 3 תווים.
 - נקודות: קיימת לפחות נקודה אחת בשם, לא קיים רצף של נקודות, התו הראשון והאחרון הם לא נקודה.
 - תקינות התווים: כלומר מסוג: אותיות באנגלית, מספרים, נקודה ומקף.
- מחזירה 1 אם השם תקין, אחרת 0.

פונקציית chek IPv4 addr:

בודקת אם מחרוזת היא כתובת IPv4 תקינה. מחזירה 0 אם המחרוזת אינה כתובת IPv4 תקינה, אחרת מחזירה ערך שונה מאפס.

פונקציית domain2DNS:

ממירה domain name לפורמט המתאים לקידוד הודעת DNS. מקבלת מחרוזת מקור ויעד, השם המומר מתעדכן המחרוזת היעד.

פונקציית DNS2domain:

ממירה domain name בפורמט המתאים לקידוד הודעת DNS לפורמט הרגיל. מקבלת מחרוזת מקור ויעד, השם המומר מתעדכן המחרוזת היעד.

פונקציית create_UDP_socket:

מקבלת מצביע ל SOCKADDR_IN כתובת IP ופורט, מחזירה סוקט. מאתחלת את Winsock, מעדכנת את SOCKADDR_IN ומייצרת את סוקט ה-UDP שיוחזר לפי הכתובת והפורט שהתקבלו ומגדירה timeout של 2 שניות עבור receive בסוקט זה.

פונקציית create_DNS_header:

מקבלת מצביע למבנה מסוג DNS_header ואת השדות הרלוונטיים ומעדכנת את השדות במבנה זה.

פונקציית create_DNS_qinfo:

מקבלת מצביע למבנה מסוג DNS_qinfo ואת השדות הרלוונטיים ומעדכנת את השדות במבנה זה.

פונקציית ntohs_DNS_header:

מקבלת מצביע למבנה מסוג DNS_header וממירה את כל השדות בו מרשת להוסט (network to host).

פונקציית ntohs_DNS_rinfo:

מקבלת מצביע למבנה מסוג DNS_rinfo וממירה את כל השדות בו מרשת להוסט (network to host).